



UNIVERSIDAD DE EXTREMADURA

Escuela Politécnica

Grado en Ingeniería Informática en Ingeniería del Software

Trabajo Fin de Grado

Análisis estadístico-gráfico de datos 3D

Pedro Alfonso Jiménez

Enero, 2019





UNIVERSIDAD DE EXTREMADURA

Escuela Politécnica

Grado en Ingeniería Informática en Ingeniería del Software

Trabajo Fin de Grado

Análisis estadístico-gráfico de datos 3D

Autor: Pedro Alfonso Jiménez

Tutor: Pablo García Rodríguez

Co-Tutor/es: Aurora Cuartero Sáez

Enero, 2019



*Dedicado a los amigos, la familia  
y, sobre todo, a cualquiera que haya  
tenido que soportarme estos años.*



## **Agradecimientos**

En primer lugar, agradecer su apoyo a los directores que han guiado este proyecto por ir más allá de su obligación e implicarse personalmente, incluyendo domingos y festivos, no me olvido, sin vuestra ayuda esto no habría sido posible, Pablo García Rodríguez y Aurora Cuartero Sáez.

También a Antonio Gordillo, quien ha participado en la etapa de pruebas de este proyecto, tu aporte nos ha sido de gran ayuda.

Y, por supuesto, a José Antonio Sánchez Sobrino, que desde el Instituto Geográfico Nacional nos ha ayudado a comprender conceptos sin los cuales no habríamos podido realizar el trabajo.





## Resumen

El presente proyecto trata del desarrollo de una librería software para el procesamiento de datos masivos, con objeto de obtener datos estadísticos y, sobre todo, su representación tridimensional.

Actualmente se encuentran multitud de datos expuestos de forma pública, cuyo potencial puede ser exprimido en proyectos de investigación, no obstante, se necesita de herramientas modernas de procesamiento automático de datos para que esta tarea sea eficiente y viable.

Con tal motivación, se presenta este proyecto, teniendo como principal objetivo facilitar el acceso al procesamiento automático de datos al público general, por lo que se ha desarrollado en un lenguaje de programación común a distintas disciplinas y de fácil manejo: Python. Del mismo modo, el código fuente se encuentra publicado en un repositorio público con licencia Apache que permite al usuario del software de la libertad de usarlo para cualquier propósito, modificarlo y distribuir versiones modificadas del software

A nivel de implementación del software se detallan los conceptos y algoritmos más importantes, la arquitectura de capas utilizada y las librerías externas que han solucionado algunos de los problemas a los que nos hemos enfrentado, principalmente los de representación tridimensional.

Como primer caso de estudio práctico, a lo largo del desarrollo de este proyecto hemos necesitado un banco de datos de suficiente volumen compuesto de coordenadas tridimensionales. Para esto, se ha utilizado un *set* ofrecido por Antonio Gordillo Guerrero. Se trata de un banco de datos sintético que simula puntos sobre superficies de superconductores.

Como segundo caso de estudio práctico, se presenta junto con la librería una ejecución basada en datos geodésicos de la red de EUREF, una red de transmisores permanentes GNSS activa a nivel europeo cuyo objetivo es permitir el almacén y consulta de datos de posicionamiento global de distintas estaciones a lo largo del tiempo. Estos datos se encuentran en un formato de intercambio distinto de las coordenadas habituales, por lo que deben procesarse en primer lugar para obtener el formato habitual (X, Y, Z).

En torno a este segundo caso de estudio, se presentan resultados obtenidos con el software sintetizados en tablas con los datos obtenidos por el software, esto incluye además distintas representaciones visuales de los datos extraídos de las mentadas estaciones GNSS, que se contextualizan sobre un mapa de la Península Ibérica.

El proyecto toma como punto de partida la librería “*VectStatsGraph3D*” desarrollada para *R*, se trata de ampliar y optimizar su comportamiento. Del mismo modo, nos apoyaremos en la Ingeniería de Geodesia para realizar un caso práctico, para lo que además se recurre a la estadística esférica con objetivo de poder dotar de sentido a los datos extraídos.

Librería se publica en el repositorio de *Python: PYPI* habiendo superado los objetivos propuestos al inicio del desarrollo, incluyendo las ampliaciones sobre la antigua librería y la mejora de su rendimiento.

# Índice de contenido

Capítulo 1: Introducción.....	1
1.1 Estructura.....	3
Capítulo 2: Estado del arte.....	5
2.1 R y VectStatsGraph3D.....	5
2.2 Geodesia y redes geodésicas.....	8
2.2.1 Datos.....	9
2.3 Estadística esférica.....	11
Capítulo 3: Objetivos.....	13
Capítulo 4: Stack tecnológico, materiales y métodos.....	15
4.1 Python.....	15
4.2 Librerías.....	15
4.3 VectStatsGraph3D en Python.....	16
4.3.1 Introducción de muestras.....	16
4.3.2 Estadística esférica.....	16
4.3.3 Representación gráfica.....	20
4.3.4 Exportar gráficos.....	20
4.4 Distribución.....	21
4.5 Caso práctico.....	21
Capítulo 5: Implementación y desarrollo.....	23
5.1 Cronología.....	23
5.2 Diagramas de casos de uso.....	25
5.2.1 Carga de datos.....	25
5.2.2 Gráficos.....	26
5.3 Diseño del sistema.....	29
5.3.1 Capa mánager.....	30
5.3.2 Capa útil.....	33
Capítulo 6: Resultados.....	35
6.1 Gráfico georreferenciado.....	57
Capítulo 7: Conclusiones y líneas futuras.....	59
7.1 Conclusiones generales.....	59
7.2 Líneas futuras.....	59
Anexo(i): Manual de usuario.....	61
Anexo(ii): Interpretación de coordenadas ERTF.....	63
Bibliografía.....	67
Páginas web y otros recursos.....	67
Referencias bibliográficas.....	67

## Índice de ecuaciones

Ecuación 1: Fórmula de la media aritmética.....	17
Ecuación 2: Fórmula de la varianza del módulo.....	17
Ecuación 3: Fórmula de la desviación estándar.....	17
Ecuación 4: Fórmula de la varianza poblacional.....	18
Ecuación 5: Fórmula de la desviación estándar poblacional.....	18
Ecuación 6: Fórmula del coeficiente de oblicuidad.....	18
Ecuación 7: Fórmula del coeficiente de Kurtosis.....	18
Ecuación 8: Fórmula de los valores para la dirección media.....	19
Ecuación 9: Fórmula de los valores medios para las coordenadas.....	19
Ecuación 10: Fórmula del módulo medio.....	20
Ecuación 11: Fórmula del error estándar esférico.....	20
Ecuación 12: Fórmula del parámetro de concentración.....	20
Ecuación 13: Fórmula para el cálculo del kde.....	27
Ecuación 14: Fórmula para el cálculo de la coordenada a partir de los residuos y la velocidad.....	64

## Índice de figuras

Figura 1: Esquema de conceptos del proyecto.....	2
Figura 2: Ejemplo de gráfico de posición de vectores TFE de Juan Carlos Ruiz Cuetos.....	6
Figura 3: Ejemplo de gráfico densidad de TFE de Juan Carlos Ruiz Cuetos.....	7
Figura 4: Ejemplo de gráfico de distribución de ángulos y módulos TFE de Juan Carlos Ruiz Cuetos.....	7
Figura 5: Red de estaciones permanentes ERGNSS [6].....	9
Figura 6: Red de estaciones permanentes EUREF [6].....	10
Figura 7: Aspecto del fichero de datos de entrada.....	16
Figura 8: Representación valores Kurtosis [8].....	19
Figura 9: Diagrama de Gantt.....	25
Figura 10: Diagrama de flujo para carga de datos.....	25
Figura 11: Tabla de vectores.....	26
Figura 12: Diagrama de flujo para gráfico de densidad.....	26
Figura 13: Ejemplo de gráfico de distribución de módulos y ángulos.....	27
Figura 14: Ejemplo de gráfico de densidad.....	28
Figura 15: Ejemplo de gráfico de posición de vectores.....	29
Figura 16: Diagrama de capa de <i>managers</i> .....	30
Figura 17: Diagrama de capa útil.....	33
Figura 18: Estación GNSS de Cáceres en la Escuela Politécnica.....	37
Figura 19: Cabecera de tabla de resultados de la estación de Cáceres.....	38
Figura 20: Cuerpo de tabla de resultados de la estación de Cáceres.....	39
Figura 21: Gráfica bidimensional de posición temporal de la estación de Cáceres.....	40
Figura 22: Gráfico georreferenciado.....	58
Figura 23: Formato fichero ERTF.....	63
Figura 24: Formato velocidades anuales ERTF.....	63
Figura 25: Formato valores residuales de estaciones ERTF.....	64

## Índice de tablas

Tabla 1: Estaciones EUREF en España.....	36
Tabla 2: Análisis de la estación de Cáceres.....	42
Tabla 3: Análisis de la estación de Sonseca.....	44
Tabla 4: Análisis de la estación de Huelva.....	46
Tabla 5: Análisis de la estación de Vigo.....	48
Tabla 6: Análisis de la estación de Almería.....	50
Tabla 7: Análisis de la estación de Valencia.....	52
Tabla 8: Análisis de la estación de Zaragoza.....	54
Tabla 9: Análisis de la estación de Santiago.....	56

# Capítulo 1: Introducción

Este proyecto nace de la necesidad de actualizar la librería “*VectStatsGraph3D*” en su versión para *R*. Se trata de una librería de análisis estadístico capaz de generar gráficos interactivos en tres dimensiones.

Para conseguir este objetivo se cuenta con el lenguaje de programación *Python*, rico en librerías de análisis estadístico y generación de gráficos, resultando además un software significativamente más rápido que los desarrollados con *R*, debido a sus características de ejecución multihilo inexistentes en *R*.

Los datos que consumirá esta librería serán tridimensionales, es decir, se analizarán puntos con valores en latitud, longitud y altura, que serán representados en forma vectorial, y pasarán por un proceso de análisis estadístico esférico que arrojará gran cantidad de resultados tanto numéricos como gráficos.

En la actualidad podemos encontrar diversas fuentes de datos que cumplan esta premisa, pero en este caso se ha optado por el análisis de datos geodésicos que son expuestos por el Instituto Geográfico Nacional (IGN) en gran cantidad. La forma que tiene el IGN de exponer estos datos es a través del siguiente *ftp*: <ftp://epncb.oma.be/epncb/station/coord>, que es alimentado por dispositivos *IoT* (*Internet of Things*, es decir, dispositivos con conexión a Internet, que además de hacer una función explícita, se comunican con el resto del mundo a través de una conexión a Internet).

Básicamente son dispositivos GPS conectados a la red que emiten su posición periódicamente con una alta frecuencia, de modo que cuando se accede a este servicio proporcionado por el ente público, se encuentra una auténtica base de datos que volumétricamente podría considerarse como *Big Data*, a la espera de que se explote adecuadamente con alguna herramienta de análisis automático como la que se desarrolla en el presente proyecto.

Para el desarrollo de esta herramienta ha sido necesario adquirir conocimiento básico sobre el software *R*, puesto que este proyecto se basa en la librería “*VectStatsGraph3D*”, desarrollada para tal lenguaje de programación, la estadística esférica, y conceptos básicos de posicionamiento GNSS y geodesia así como sus formatos, además del lenguaje *Python* en el que ha sido desarrollado como se representa en la figura 1.

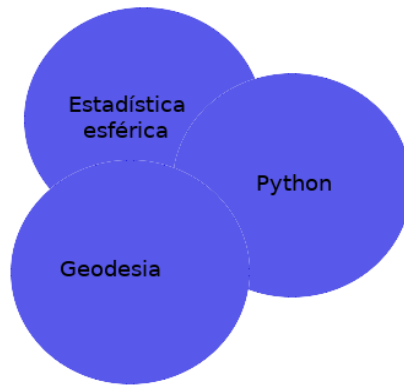


Figura 1: Esquema de conceptos del proyecto

En líneas generales, el proyecto trata del uso de la estadística esférica para analizar los datos emitidos por estaciones GNSS. El valor de este proyecto se basa en ofrecer una solución de funcionamiento rápido desarrollada en el lenguaje de programación *Python*, además se consigue dotar de significado visual a los estadísticos esféricos calculados para los datos GPS, permitiendo así una interpretación más natural de los resultados obtenidos, tal como podemos ver en el gráfico georreferenciado de toda la Península Ibérica, más adelante en este mismo documento en la figura 22 (página 50).

Se entiende que esto es una propuesta de valor importante, debido a que el IGN, así como otras organizaciones europeas, ofrecen estos datos en bruto, resultando en conjuntos de muestras masivas, que pueden ser procesadas con otro software, para obtener los estadísticos lineales o numéricos, pero no se encontraba previamente una solución que permitiera la representación gráfica tridimensional.



## 1.1 Estructura

A continuación se expone la estructura del documento como apoyo a su lectura para facilitar la búsqueda de información y situación en las distintas secciones del mismo:

- **Estado del arte:** Donde se encuentra el estado previo al desarrollo de este proyecto y un esquema sobre la librería "VectStatsGraph3D" previamente mencionada. Esto incluye una introducción a la geodesia y redes geodésicas, así como al formato de los datos distribuidos por la red objetivo de estudio del proyecto y una introducción a la estadística esférica.
- **Objetivos:** La propuesta de objetivos del proyecto y su alcance.
- **Stack tecnológico, materiales y métodos:** Donde se detalla el lenguaje de programación, las plataformas utilizadas y los requisitos funcionales en profundidad.
- **Implementación y desarrollo:** Justificación de decisiones de desarrollo, así como el detalle de las cuestiones más importantes del mismo y el tiempo invertido, incluyendo un diagrama de Gantt.
- **Resultados:** Donde se exponen los resultados de la prueba realizada sobre algunas de las estaciones pertenecientes a la red de estaciones GNSS EUREF y la discusión sobre estos.
- **Conclusiones:** Conclusiones finales del proyecto y líneas futuras.



## Capítulo 2: Estado del arte

En la realización de este proyecto encontraremos un compendio de conceptos recurrentes: geodesia, análisis estadístico esférico tanto numérico como gráfico y el conjunto de tecnologías existentes para automatizar el trabajo en estos campos. Para comprender correctamente el estado previo al desarrollo de la memoria del presente proyecto en estas doctrinas, se desarrolla en los siguientes párrafos el estado del arte de los mismos.

Si pensamos en el procesamiento de datos con fines estadísticos, y sobre todo, en su representación tridimensional, actualmente es habitual tener que recurrir a software especializado, normalmente privativo, y desarrollado para plataformas y en lenguajes poco inclusivos para personas que encuentran su campo profesional lejos de la Estadística, ejemplos de esto son *SPSS*, *XLSTAT*, *MATLAB* o, en el campo del desarrollo de software para este fin *R*, de donde se parte para la realización del presente proyecto, debido a que se va a trabajar sobre la librería “*VecStatsGraph3D*” desarrollada en este lenguaje.

Además se desarrolla una prueba de funcionamiento de la librería basada en datos geodésicos, para lo que será necesario comprender esta disciplina, así como tener acceso a datos tridimensionales que puedan interpretarse en términos geodésicos. Para esto, se ha hecho uso del conocimiento adquirido en el campo de la estadística esférica, que permitirá el análisis numérico de las muestras de datos obtenidas y aún más importante para los objetivos del proyecto, el análisis gráfico basado en este tipo de estadística, que se espera, ayude a la interpretación y comprensión de los resultados.

A continuación se profundiza en cada uno de los puntos considerados clave, de las disciplinas comentadas anteriormente. Se espera que puedan ayudar al lector, a tener un punto de partida básico para comprender el resto del documento.

### 2.1 R y VectStatsGraph3D

“R” es un lenguaje de programación orientado al análisis gráfico y estadístico: <https://www.r-project.org/> hasta el momento es el software más utilizado para resolver problemas relacionados con estas disciplinas, particularmente entre desarrolladores ajenos a la Ingeniería de Software, aún con esto, es un lenguaje presentado al mercado en 1993, que no ha evolucionado mucho y arrastra algunos problemas que trataremos más adelante en este documento.

En el repositorio público de librerías de *R*, “*cran*”, se puede encontrar la librería “*VectStatsGraph3D*”: <https://cran.r-project.org/web/packages/VecStatGraphs3D/> desarrollada en el año 2010 como Trabajo de Fin de Estudios para la Universidad de Extremadura por Juan Carlos Ruiz Cuetos [1], en este caso se trata de una librería

abierta, con licencia GPL y fácilmente accesible, se encuentra desarrollada en *R* y su última actualización data de 2014.

La herramienta tiene como objetivo el análisis estadístico tanto numérico como gráfico de una muestra de vectores tridimensionales.

A nivel numérico analiza los módulos de los vectores en términos de estadística lineal, y sus ángulos y orientaciones por estadística esférica.

Además, puede construir ciertas gráficas tridimensionales para facilitar la interpretación de los cálculos obtenidos sobre la entrada de datos como se muestra a continuación.

En “VecStatsGraph3D” para *R* encontramos la posibilidad de realizar gráficos de posicionamiento de vectores, donde, como se observa en la figura 2, existen ciertos problemas de representación, probablemente debido a la desproporción entre las magnitudes de los tres ejes, o tal como comenta Juan Carlos Ruiz Cuetos en la sección de líneas futuras de su documentación.

“-Mejorar la creación del gráfico de Vectores 3D con muchos datos de entrada (si la cantidad de datos es muy grande, puede llegar a colapsar la máquina de trabajo)”.

”-Mejorar la velocidad de representación de las flechas del gráfico de Distribución de módulos y ángulos. Una forma de mejorar esto, puede ser, dibujar el conjunto de todas las flechas a la vez, en lugar de una a una como se realiza en esta versión, aprovechando la potencia de cálculos vectoriales y matriciales de *R*” [1].

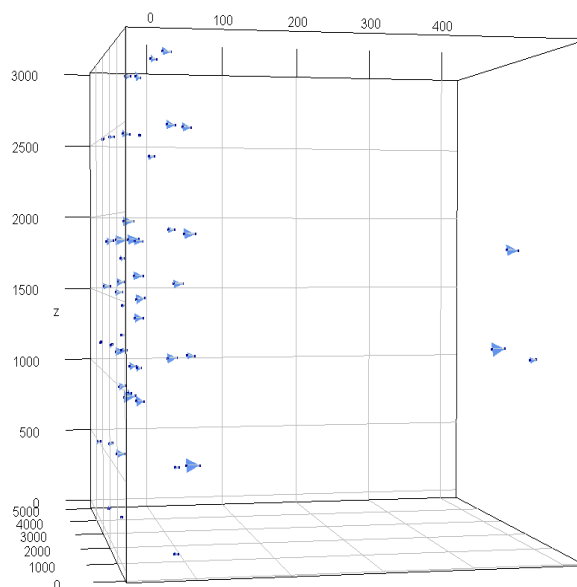


Figura 2: Ejemplo de gráfico de posición de vectores TFE de Juan Carlos Ruiz Cuetos

En la figura 3 se pueden observar las superficies representando las posiciones más frecuentes de los puntos de la muestra analizada, como se observa está basado en el dibujo de superficies con un código de colores, representando zonas con más cantidad de población los tonos más vivos, siendo zonas con menos probabilidad de encontrar puntos los tonos grisáceos.

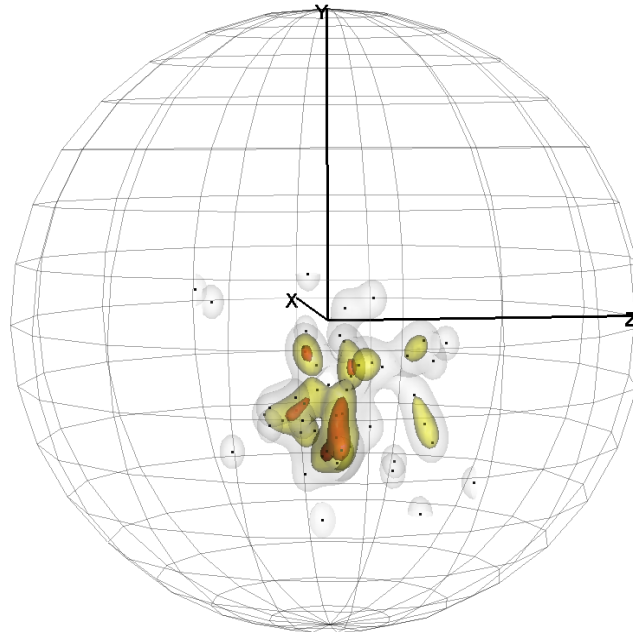


Figura 3: Ejemplo de gráfico densidad de TFE de Juan Carlos Ruiz Cuetos

Del mismo modo, es posible dibujar un gráfico de representación de módulos y ángulos, tal como se muestra en la figura 4.

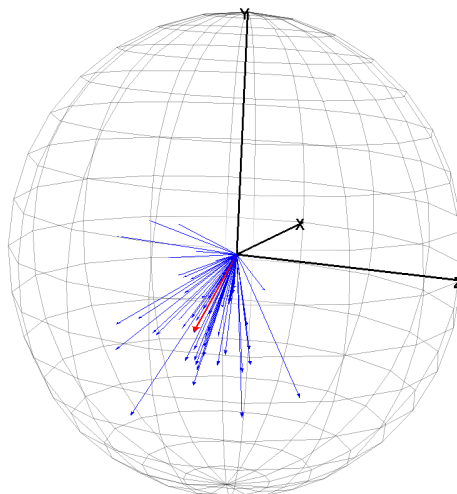


Figura 4: Ejemplo de gráfico de distribución de ángulos y módulos TFE de Juan Carlos Ruiz Cuetos

Funcionalmente, esta librería se integra con un *IDE* de *R*, se ejecuta de forma aislada y

mediante la consola de comandos se introduce la ruta del fichero con las muestras de datos, se realizan llamadas a las funciones para realizar cálculos y mostrar gráficas. La gran mayoría de los cálculos realizados por la librería han sido desarrollados y se encuentran en código abierto dentro de la librería, lo que facilita su comprensión. Sin embargo, *R* es una implementación del antiguo software *S* que funciona con un solo hilo, por lo que tiene una velocidad de funcionamiento lenta.

## 2.2 Geodesia y redes geodésicas

*“La Geodesia es la ciencia que estudia la forma y dimensiones de la Tierra. Esto incluye la determinación del campo gravitatorio externo de la Tierra y la superficie del fondo oceánico. Dentro de esta definición, se incluye también la orientación y posición de la tierra en el espacio” [5].*

Según esto, se extrae que también se trata de procedimiento ingenieril consistente en resolver el problema de la geolocalización global de posiciones terrestres, basado en el uso de satélites para obtener posiciones, en forma de coordenadas, de puntos sobre la superficie de la Tierra teniendo en cuenta la altura a la que se encuentran, de tal modo que puedan ser consultados con independencia del momento temporal en que se tomaron las muestras. Aunque esto tiene multitud de aplicaciones prácticas, el presente proyecto se va a centrar en el muestreo de datos de estaciones permanentes que emiten su posición de forma periódica, con lo que se puede estudiar como evoluciona la situación real de ciertos puntos del espacio a lo largo del tiempo. Es lo que conocemos como **Geodesia espacial**: “Determinación de coordenadas a partir de mediciones efectuadas a satélites artificiales u otros objetos naturales o artificiales exteriores a la Tierra (GNSS, VLBI, SLR, DORIS) y relación con la definición de sistemas de referencia” [2].

Con dicho fin, se establece el concepto de **red geodésica**: un conjunto de estaciones GPS ubicadas en puntos con coordenadas estables conocidas con objeto de emitir, a lo largo del tiempo las variaciones de posicionamiento en latitud, longitud y altura, pudiendo obtener así la posición diferencial de estos puntos en periodos temporales.

Un ejemplo de red geodésica, que se desarrolla en este documento es la red ERGNSS, que está formada por unas 80 estaciones GNSS distribuidas en el territorio español de las que 24 pertenecen también al sistema europeo EUREF, como puede observarse en la figura 4.

Estas estaciones son monumentos equipados con receptores y emisores GPS que permiten dos tipos de servicios:

- **Servicio de post-proceso**: La red de estaciones dispone públicamente los datos emitidos en diferentes intervalos de tiempo. Estos datos son almacenados en un formato de intercambio internacional conocido como **RINEX** que tiene ciertas peculiaridades, se analiza en el siguiente apartado de la documentación.

El presente proyecto se apoya en este servicio para realizar algunas pruebas de funcionamiento del software.

- **Servicio de posicionamiento en tiempo real:** Las estaciones ofrecen un servicio *en streaming* de los datos a medida que se van emitiendo. Este servicio no se encuentra actualmente en funcionamiento, al menos en el enlace proporcionado por el Instituto de Geodesia Nacional, en adelante "IGN".



Figura 5: Red de estaciones permanentes ERGNSS [6]

### 2.2.1 Datos

Como se comenta previamente, para la ejecución de una prueba de campo de esta librería se van a utilizar los datos históricos de la red ERGNSS, concretamente los emitidos por las estaciones pertenecientes también a la red EUREF.

Estos datos se encuentran públicos en formato RINEX, que es un formato de intercambio internacional de datos GPS. Está formado por tres tipos de ficheros: de observaciones, de navegación y de datos meteorológicos. Estos permiten obtener información extra además del posicionamiento de la estación, aunque para la realización de la prueba que se menciona, bastará con el manejo del fichero de observaciones que es el tratado en los párrafos siguientes.

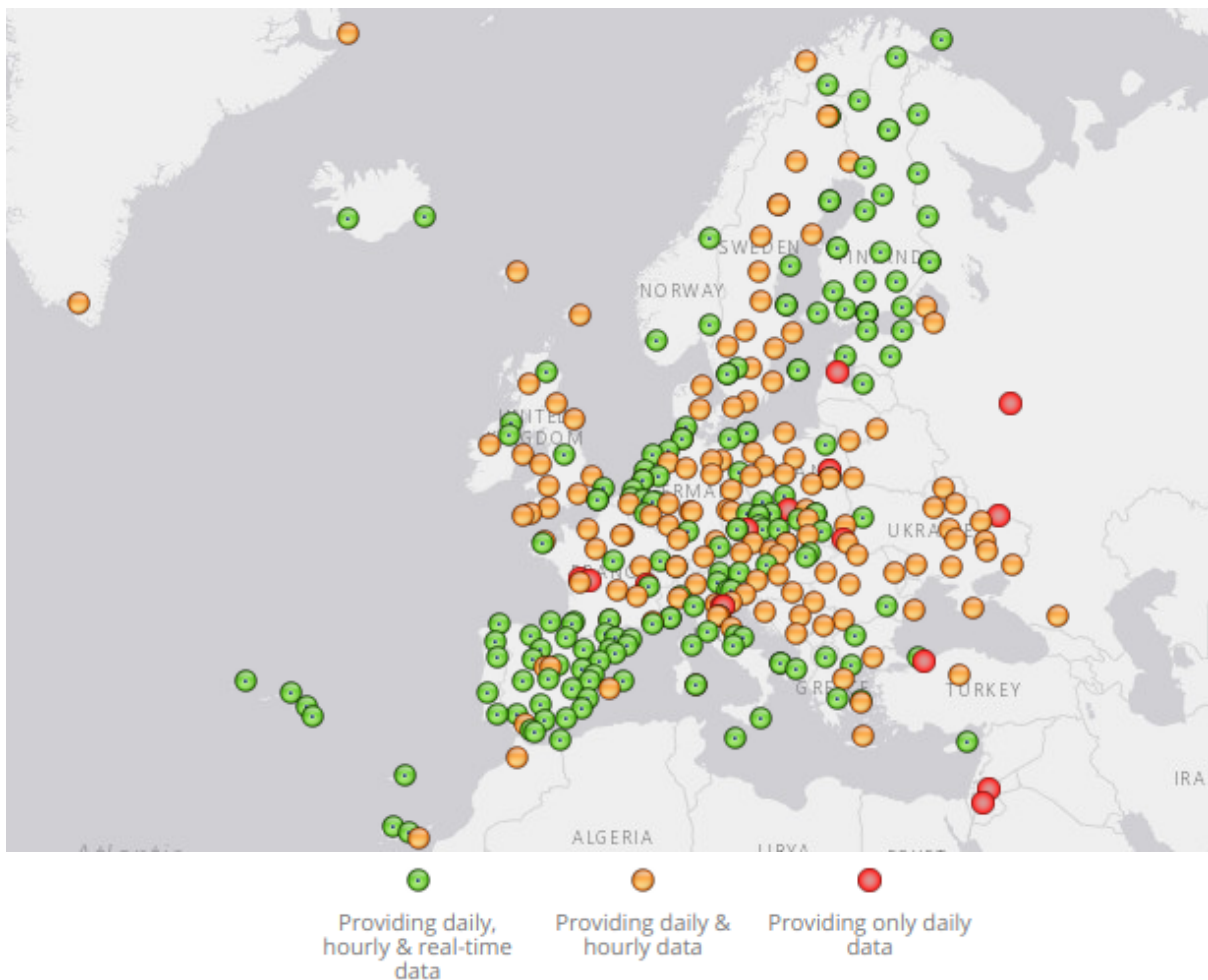


Figura 6: Red de estaciones permanentes EUREF [6]

En la figura 6 se muestran las estaciones permanentes de esta red, en rojo las que sólo proveen datos diarios, en amarillo las que los proveen diaria y horariamente, y en verde las que además las emiten datos en tiempo real.

Para identificar las coordenadas de cada una de estas estaciones, así como otros datos necesarios (época, velocidad de la estación... y otros conceptos que se desarrollan en el Anexo ii) existe un fichero global en el que se pueden consultar accesible desde este ftp: [ftp://epncb.oma.be/epncb/station/coord/EPN/EPN\\_A\\_ETRF2014.SSC](ftp://epncb.oma.be/epncb/station/coord/EPN/EPN_A_ETRF2014.SSC)

Además, cada estación tiene su propio fichero de datos residuales, donde se encuentra la desviación en cada dirección: Norte-Sur, Este-Oeste y altura a lo largo del tiempo. Como ejemplo, en el siguiente enlace se puede acceder a la tabla de residuales de la estación de Cáceres:

[ftp://epncb.oma.be/pub/station/coord/EPN/Time\\_Series/CACE\\_13447M001.dat](ftp://epncb.oma.be/pub/station/coord/EPN/Time_Series/CACE_13447M001.dat)

Para dotar de significado a los datos obtenidos de esta red de estaciones, será necesario cierto conocimiento de estadística esférica, que se introduce en el punto siguiente.



## 2.3 Estadística esférica

Así como la estadística lineal se aplica a datos unidimensionales, y la circular a datos de dos dimensiones, la estadística esférica es aplicable a datos tridimensionales, en formato (X, Y, Z). Es por ello que el uso de esta disciplina será necesario para dar significado a los datos extraídos de nuestras coordenadas tridimensionales.

Este tipo de estadística es más compleja que las anteriores, al incluir una dimensión más que la circular, y a menudo es difícilmente interpretable en su forma estrictamente numérica, es por eso, que en este proyecto se facilitará la interpretación gráfica de algunos estadísticos calculados.

En el siguiente apartado “Objetivos” se reflejan los estadísticos desarrollados como alcance de la aplicación, todos ellos producto de la evolución del trabajo de Bachelet (1981) en su libro “Circular statistics in biology”[2] y Fisher en “Statistical analysis of circular data”[3], cuyo contenido se entiende como la base de la estadística circular aplicada, que es extrapolable a la mencionada estadística esférica también desarrollada en “Statistical Analysis of Spherical Data” por Fisher, Lewis y Embleton [4].



## Capítulo 3: Objetivos

Aprovechando las facilidades que ofrece *Python* como lenguaje de programación, se pretende publicar una librería con las mismas capacidades que encontramos en la versión *R* distribuida de forma libre en su repositorio de librerías *PyPI*, mejorando tanto su representación gráfica como su rendimiento. Esto será, cumplir con los requisitos funcionales desarrollados en el Capítulo 4, que se pueden agrupar en tres puntos:

- 1. Estadísticos numéricos:** Divididos en dos grupos; los extraíbles de los módulos de los vectores y los que se deducen de la posición y el ángulo que forman.
- 2. Representaciones gráficas:** Tres representaciones gráficas completamente interactivas, o escenas, exportables en formato imagen. Representan mapas de densidad, gráfico de distribución de vectores y ángulos, y gráfico de posición de vectores.
- 3. Mecanismo de consumo de datos:** Un mecanismo de consumo de datos, que actualmente se basa en leer datos de un fichero de texto, y en el caso de los extraídos de la red EUREF, transformarlos a un formato procesable por la librería.

Y los requisitos no funcionales, también descritos en el capítulo siguiente, tales como el rendimiento de la aplicación y su facilidad de acceso.

En cuanto a las ventajas que justifican la elección de este lenguaje para la modernización de la librería, caben señalar los siguientes párrafos:

*Python* no es sólo más potente que *R* a nivel de rendimiento, sino que también goza de una sintaxis mucho más liviana haciendo que su curva de aprendizaje sea mucho menos pronunciada y, especialmente para profesionales ajenos al sector de la estadística, significativamente más corta. Esto es algo importante debido a que la interpretación estadística es cada vez más frecuente en otros sectores, principalmente en el de la Ingeniería del Software, de este modo, se espera que sea muy sencillo instalar esta librería en cualquier aplicación que se esté desarrollando con este lenguaje y utilizarla de forma sencilla.

Por otra parte, tal como se menciona previamente, se distribuirá con una licencia libre para permitir que la comunidad la mejore y amplíe en función de sus necesidades, lo que resulta particularmente atractivo observando la gran cantidad de usuarios que colaboran en esta comunidad.

Se espera que la librería sea capaz de consumir datos de la red geodésica EUREF para realizar un estudio sobre la tendencia de movimientos de alguna de sus estaciones. Se han elegido con criterio de estar situadas en territorio español y distribuidas por la periferia y el centro de la Península Ibérica con objetivo de encontrar alguna relación entre la variación de sus movimientos.

Debido a la naturaleza científica de los trabajos en los que se utilice esta librería, también se podrá exportar en alta resolución las representaciones gráficas generadas con el software. La intención de este requisito es poder trasladar no sólo los resultados numéricos, sino también los visuales a artículos de investigación con calidad suficiente.

## Capítulo 4: Stack tecnológico, materiales y métodos

Se detallan a continuación el lenguaje de programación utilizado para el desarrollo de este proyecto, las librerías externas que han resuelto los problemas principales de representación gráfica y análisis matemático, además del detalle de los requisitos funcionales y no funcionales con los que cumple el proyecto.

### 4.1 Python

Se decide utilizar Python como lenguaje para desarrollar este proyecto debido a las facilidades que ofrece en cuanto a librerías y *frameworks* externos para el procesamiento de operaciones matemáticas y representación gráfica de los mismos.

Además se espera que su rendimiento sea muy superior a la versión desarrollada en *R*, principalmente debido a la naturaleza multi-hilo de *Python*.

Aunque la primera decisión fue construir la librería con la versión 3 Python, se decidió volver a la versión 2.7 al comprobar que las librerías utilizadas para la representación y navegación por los gráficos estaban dando problemas con las versiones más modernas de este lenguaje.

### 4.2 Librerías

Se detallan a continuación las librerías de terceros más significativas que se han utilizado para el desarrollo de este proyecto:

- **NumPy:** Se trata de una librería para el cálculo científico capaz de simplificar la apariencia del código, condensando las operaciones algebraicas básicas en funciones muy simples y de fácil manejo. Además opera con *arrays* de forma paralela, utilizando diferentes hilos para cada valor contenido en esta estructura lo que supone un gran aumento de rendimiento, incluso comparándolo con otras soluciones como la de *R* (<http://www.numpy.org/>).

El uso de esta librería ha resultado básico para cumplir con los objetivos de rendimiento de la aplicación, se recomienda encarecidamente que en cualquier modificación futura del comportamiento de la aplicación se continúe utilizando esta solución para la resolución de cálculos matemáticos.

- **SciPy:** Es otra librería para el cálculo científico y algebraico que complementa con sus funciones algunos algoritmos que no se encontraban en *NumPy*.
- **Matplotlib:** Es la librería utilizada para la representación gráfica de los datos, permite de forma muy sencilla representar escenas tridimensionales, incluyendo diferentes objetos dentro de ellas, como vectores, curvas, superficies ... además

permite la navegación en la representación, esto es, zoom, rotaciones y traslaciones.

Por último encontramos la posibilidad de exportar una imagen de la escena en alta resolución lo que, será útil para que el usuario de esta librería imprima los resultados de su investigación, pudiendo compartirlos con el resto de su comunidad en un formato adecuado.

### 4.3 VectStatsGraph3D en Python

A continuación se detalla el alcance de los requisitos funcionales cubiertos por la nueva librería en Python, y los no funcionales cubiertos por el proyecto.

#### 4.3.1 Introducción de muestras

Los datos serán leídos de un fichero en una ruta introducida por el usuario, su formato deberá adaptarse a coordenadas tridimensionales (x, y, z) para el correcto funcionamiento de los algoritmos desarrollados en esta librería.

En cada línea del fichero deberán encontrarse seis valores numéricos separados por uno o más espacios, esto representará dos grupos de valores, los tres primeros serán el origen del vector y los siguientes el punto de destino en el espacio.

De este modo, cada línea del fichero de entrada representa un vector. Los valores decimales serán representados con un punto (.) separando la parte entera de la fraccionaria del valor, tal como se muestra en la figura 7.

```
-59.0000      4814.0000      2545.0000      -57.7014      4814.4934      2546.4644
-61.0000      4969.0000      1118.0000      -60.7520      4967.0451      1125.7559
-66.0000      4907.0000      423.0000      -62.2877      4912.4627      426.0798
-61.0000      4466.0000      1522.0000      -52.8442      4462.6964      1524.0323
-56.0000      4089.0000      2539.0000      -50.6367      4090.2784      2540.9400
-59.0000      4212.0000      1830.0000      -51.2981      4210.1975      1837.2748
```

Figura 7: Aspecto del fichero de datos de entrada

#### 4.3.2 Estadística esférica

*“La estadística esférica trata con datos angulares tales como ejes y vectores en un espacio tridimensional. Estos datos están definidos por dos ángulos: longitud y colatitud, (también llamados ángulo cénit y azimuth, y un módulo de magnitud lineal)”.*

*El análisis estadístico de datos esféricos comenzó con Fisher, quien desarrolló una distribución para errores angulares en una esfera. Fisher investigó las propiedades de la media estadística esférica (...) En un trabajo posterior de Fisher y Embleton se desarrolló un trabajo puramente dedicado al análisis de datos estadísticos.” [7]*

A continuación se detallan las funciones estadísticas que se pretende que cubra la librería divididas en dos secciones: las extraíbles de los módulos de los vectores y las que se

obtienen de los ángulos y las posiciones.

Los estadísticos extraíbles de los módulos de los vectores:

- **Mínimo:** El valor mínimo de la muestra.
- **Máximo:** El valor máximo de la muestra.
- **Media aritmética:** Calcula la media aritmética de los elementos de la muestra.

$$\bar{x} = \frac{\sum_{i=1}^N X_i}{N}$$

Ecuación 1: Fórmula de la media aritmética

- **Varianza de módulo:** Calcula el valor de la varianza muestral a partir de la división entre el sumatorio de la diferencia al cuadrado de los módulos menos la media aritmética y el número de elementos menos 1.

$$S^2 = \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n-1}$$

Ecuación 2: Fórmula de la varianza del módulo

- **Desviación estándar:** Calcula el valor de la desviación estándar a partir de la raíz cuadrada de la división del sumatorio de la diferencia al cuadrado de los módulos menos la media aritmética, entre el número de elementos menos uno, o la raíz cuadrada de la varianza.

$$s = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2}$$

Ecuación 3: Fórmula de la desviación estándar

- **Varianza poblacional:** Calcula el valor de la varianza poblacional a partir de la división del sumatorio de la diferencia al cuadrado de los módulos menos la media aritmética, entre el número de elementos.

$$\delta^2 = \frac{\sum_{i=1}^n (X_i - \bar{x})^2}{n}$$

Ecuación 4: Fórmula de la varianza poblacional

- **Desviación estándar poblacional:** Calcula el valor de la desviación típica poblacional a partir de la raíz cuadrada de la división del sumatorio de la diferencia al cuadrado de los módulos menos la media aritmética, entre el número de elementos.

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}}$$

Ecuación 5: Fórmula de la desviación estándar poblacional

- **Coefficiente Skewness:** Mide el grado de asimetría de una distribución. Toma valor 0 cuando es totalmente simétrica, valores negativos cuando se trata de una distribución sesgada a la izquierda y valores positivos si es sesgada al lado contrario.

$$\text{Skewness} = \frac{[(N-1)(N-2)]}{N} \sum_{j=1}^N \frac{(x_j - \bar{x})^3}{\sigma_x^3}$$

Ecuación 6: Fórmula del coeficiente de oblicuidad

- **Coefficiente Kurtosis:** Indica el grado de concentración de una distribución alrededor de su moda. Si la distribución tiene un conjunto de valores bien ajustado a su moda, resultará en un valor de Kurtosis  $\approx 0$ . Si la distribución presenta en la moda un pico más elevado que el que correspondería a una distribución normal envolvente (leptocúrtica), el coeficiente resultará en valores positivos y si la moda presenta un pico menor (platicúrtica) el resultado serán valores negativos, como puede observarse en la figura 8.

$$\text{Kurtosis} = \frac{1}{n} \sum_{i=1}^n \left( \frac{x_i - \bar{x}}{SD(x)} \right)^4$$

Ecuación 7: Fórmula del coeficiente de Kurtosis



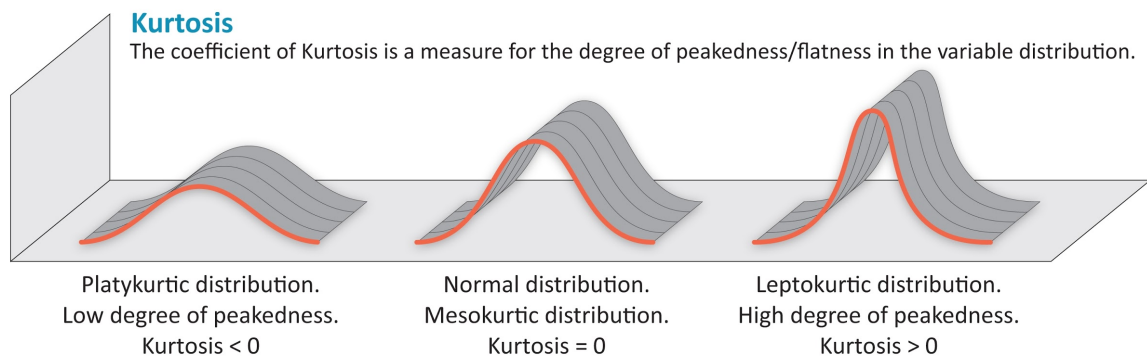


Figura 8: Representación valores Kurtosis [8]

Los estadísticos extraíbles de las posiciones y ángulos de los vectores:

- **Dirección media:** Formada por la colatitud media obtenida a partir del arcoseno de la media de las coordenadas Z, y la longitud media obtenida a partir de la arcotangente de la media de las coordenadas Y entre la media de las coordenadas X.

$$\hat{\theta} = \arccos(\hat{Z})$$

$$\hat{\phi} = \arctan\left(\frac{\hat{Y}}{\hat{X}}\right)$$

Ecuación 8: Fórmula de los valores para la dirección media

Ecuación 9: Fórmula de los valores medios para las coordenadas

- **Módulo medio:** Tomará valores oscilantes entre 0 y 1, siendo todos los puntos de la distribución coincidentes si el resultado es 1.

$$\bar{R} = \frac{R}{n}$$

#### Ecuación 10: Fórmula del módulo medio

- **Error estándar esférico:** Error estándar de la media aritmética.

$$\sqrt{\frac{\text{sum}(\text{modules} - \text{ArithmeticMean3D}(\text{modules}))^2}{\text{NumberOfElements3D}(\text{modules}) * (\text{NumberOfElements3D}(\text{modules}) - 1)}}$$

#### Ecuación 11: Fórmula del error estándar esférico

- **Parámetro de concentración:** Parámetro de concentración. Si resulta en 0, se tratará de una distribución uniforme. Cuando tiende a infinito la distribución se concentra en un punto. Este parámetro define la distribución Fisher.

$$\kappa = \frac{(n - 1)}{(n - R)}$$

#### Ecuación 12: Fórmula del parámetro de concentración

### 4.3.3 Representación gráfica

El software permite la visualización de los siguientes gráficos tridimensionales que son interactivos, siendo posible hacer zoom sobre ellos, así como rotar la vista además de generar una imagen en alta resolución para exportarla.

Estas representaciones ayudan a comprender la información que se extrae de las muestras de datos procesadas..

- **Gráfico de distribución de módulos y ángulos:** Este gráfico muestra un vector por cada muestra siendo su longitud el módulo del vector. Estos vectores partirán del origen de coordenadas [0,0,0] hasta la posición del dato de la muestra [x,y,z].
- **Mapa de densidad:** El mapa representa la densidad de los valores de las muestras, pudiendo identificar las distintas áreas en función del color, siendo los más vivos aquellas zonas con mayor densidad de población.
- **Gráfico de vectores:** Representa un gráfico de posición de los vectores en el espacio.

### 4.3.4 Exportar gráficos

El software permite activar una opción para exportar el resultado de las representaciones

gráficas en alta calidad, en formato `svg` pudiendo así adjuntarse a otros documentos en la mejor calidad posible. Esta opción se puede activar al inicio de la ejecución de la librería, cuando el software pregunta por línea de comandos al usuario si quiere almacenar la imagen exportada.

#### 4.4 Distribución

La librería se distribuye mediante el gestor de paquetes *PyPI* "<https://pypi.org/>" que es la herramienta propia de la comunidad de Python para la instalación e intercambio de librerías externas. Se ha licenciado de modo que pueda ser utilizada libremente así como modificada y ampliada de acuerdo con el gusto y las necesidades de la comunidad, según la licencia Apache que *como cualquier otra de las licencias de software libre, "Permite al usuario del software la libertad de usarlo para cualquier propósito, distribuirlo, modificarlo, y distribuir versiones modificadas de ese software" [9].*

#### 4.5 Caso práctico

Se realizará una prueba de ejecución de la librería en base a los datos extraídos del sistema EUREF, se obtendrán los estadísticos de algunas de las estaciones de la península ibérica, así como sus gráficas de distribución de módulos y ángulos, además se pretende situar estas gráficas en un mapa para facilitar la comprensión de los datos y las posiciones de las estaciones.



## Capítulo 5: Implementación y desarrollo

En este apartado se van a desarrollar las cuestiones de técnicas de implementación y desarrollo, en concreto la cronología del proyecto, los diagramas de casos de uso más importantes, así como una visión general de la arquitectura del sistema incluyendo la especificación de las funciones más importantes.

### 5.1 Cronología

El desarrollo del proyecto se divide en tres fases: de planificación, de desarrollo y de documentación, que se explican más adelante, como se puede ver en la figura 8, la planificación del proyecto se ha realizado por semanas.

Desde el inicio del trabajo se define un horario de lunes a jueves a razón de cuatro horas por día, a excepción del mes de agosto, que se tomará como vacaciones. Con esta planificación se esperaba terminar todas las tareas del proyecto a tiempo para el 23 de noviembre de 2018.

**1. Fase de planificación:** Se inicia en la tercera semana del mes de julio, y termina a finales del mismo mes. En esta fase se definen los objetivos del proyecto, y se realiza una labor de documentación previa al desarrollo, en total se invierten 12 horas de trabajo:

- Definición de objetivos: Durante esta semana se dedican esfuerzos a la definición de objetivos y planificación del trabajo general.
- Documentación: Son dos semanas previas a la implementación en exclusiva dedicadas, en parte a la labor de documentación previa de algunos conceptos, aquí se incluye el estudio de la librería *VectStatsGraph3D* para *R*, así como la lectura de artículos relacionados con ella, además del estudio de los conceptos estadísticos necesarios para el desarrollo.

**2. Fase de desarrollo:** Iniciada en la cuarta semana de julio y conviviendo con el estudio durante esta misma con la labor de documentación, se extiende hasta el final del proyecto en noviembre, siendo una labor exclusiva durante el mes de septiembre y las tres primeras semanas de octubre.

Es la fase más fuerte, debido a que aquí se incluye el desarrollo programático del proyecto, donde también se incluye el aprendizaje sobre el lenguaje de programación *Python* y una semana dedicada a la investigación y puesta en funcionamiento de la distribución de la librería en *PYPI*, además de dos semanas dedicadas a las pruebas con los datos que iban postulándole como objetivo final de las pruebas. En total se invierten unas 144 horas:

- **Implementación y desarrollo:** Es la fase de implementación programática de la librería, se trata, sin dudas, la más larga del proyecto, comenzando la segunda semana y terminando el último día del desarrollo, esto es así debido a que durante la fase de pruebas, al experimentar con distintos formatos de datos, se hizo patente la necesidad de incluir nuevas funciones que permitieran automatizar estos procesos
  - **Pruebas:** Realizada durante la primera quincena de noviembre, esta fase trata de probar la librería con los datos recogidos, tanto con los definitivos como con el resto que se detallan en la fase de documentación.
  - **Distribución:** Durante esta fase se realiza un estudio sobre el repositorio *PYPI* para distribuir la librería, se suben las primeras versiones y se prepara todo el proceso para automatizar siguientes publicaciones.
- 3. Fase de documentación:** Iniciada en la última semana de octubre, y concluyendo con la finalización del proyecto, se dedica este periodo a la búsqueda de datos con los que probar la librería, así como al desarrollo del presente documento. Tiene una duración total de 40 horas.
- **Búsqueda de datos:** Desde la última semana de octubre hasta la segunda de noviembre, se dedican esfuerzos a la búsqueda de datos que doten de significado a algún ejemplo que sirva para probar la librería. En los primeros momentos de esta fase se utilizan *sets* de datos sintéticos, y se va iterando en torno a distintas opciones, finalmente, como se detalla en otros puntos de este documento, se prueba con simulaciones de superficies de superconductores, aunque el ejemplo real de funcionamiento es realizado en torno a los datos geodésicos previamente descritos.
  - **Documentación del proyecto:** La segunda fase más larga del proyecto, toma cuatro semanas de trabajo y trata de la confección de este documento.

En total, el trabajo realizado para desarrollar este proyecto ronda las 196 horas, distribuidas en las fases descritas anteriormente, como se puede observar en la figura 9.

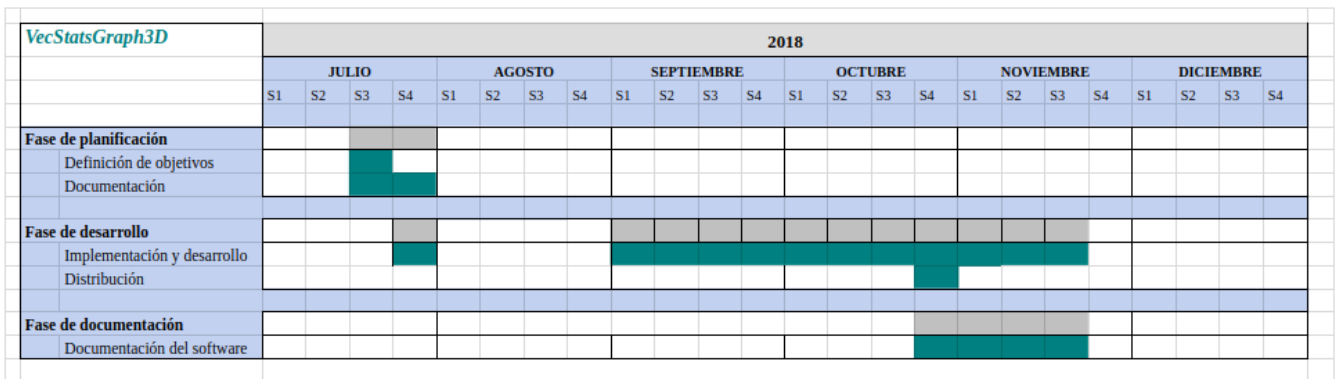


Figura 9: Diagrama de Gantt

## 5.2 Diagramas de casos de uso

A continuación se detallan los casos de uso de los algoritmos más importantes que son principalmente, los generadores de gráficos y la carga de datos. Para las tres escenas generables el diagrama es el mismo: el sistema es alimentado con los datos mediante un fichero de texto formateado correctamente, el sistema realiza los cálculos necesarios y devuelve la representación gráfica al usuario. Lo único que cambia para cada gráfico es el cálculo de los estadísticos necesarios en el módulo de gráficos. Se detallan a continuación.

### 5.2.1 Carga de datos

Como se observa en la figura 10 el usuario proporciona al módulo gestor de ficheros la ruta del archivo donde se encuentran sus muestras de datos, tras esto, haciendo uso del módulo “*Vector util*” se realizan las operaciones necesarias para crear la estructura de datos con la que trabajará el sistema.

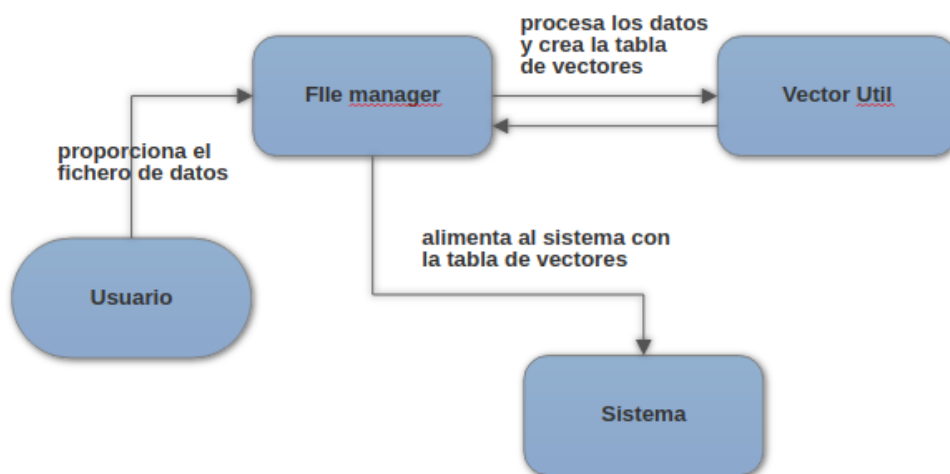


Figura 10: Diagrama de flujo para carga de datos

Esta estructura es una tabla que representa la información sobre los vectores, uno por cada línea, como se muestra a continuación.

Módulo	Colatitud	Longitud	X	Y	Z	X <sub>i</sub>	Y <sub>i</sub>	Z <sub>i</sub>	X <sub>f</sub>	Y <sub>f</sub>	Z <sub>f</sub>
.	.	.	.	.	.	.	.	.	.	.	.

Figura 11: Tabla de vectores

En la figura 11 se muestra la estructura de datos con la que trabaja el sistema una vez procesados por el módulo de carga de ficheros. Se trata de una matriz de  $n \times m$  elementos, siendo  $m$  el número de vectores que se encuentran en la muestra de datos original y  $n$  las columnas que se observan en la figura: módulo, colatitud, longitud, incremento de coordenada X, incremento de coordenada Y e incremento de coordenada Z. Las seis siguientes celdas corresponden a los valores de origen y final del vector.

### 5.2.2 Gráficos

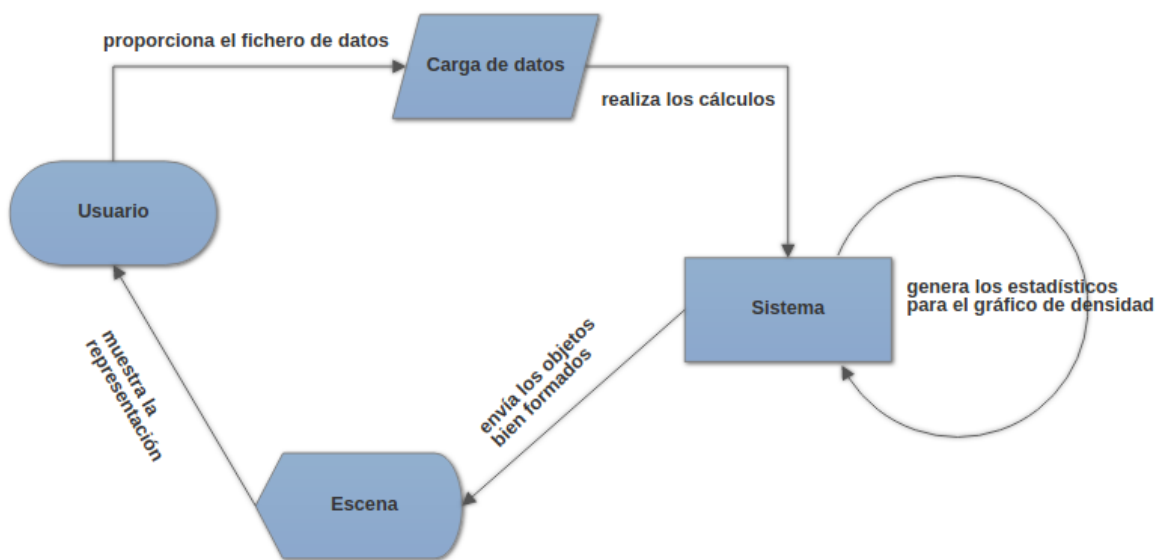


Figura 12: Diagrama de flujo para gráfico de densidad

En la figura 12 se muestra el diagrama de flujo para los algoritmos de generación de gráficos, que se detallan en particular más adelante. La primera parte del diagrama, la interacción entre el usuario y el módulo de carga de datos funciona tal como se explica en el caso de uso anterior, la tabla de salida de este módulo se envía al sistema, que dependiendo del gráfico que se quiera generar será el *mánager* del gráfico de distribución de ángulos, de densidad o de posición de vectores.

Los gráficos que se muestran a continuación como prueba de funcionamiento de estos gráficos, corresponden con unas simulaciones de superficies superconductoras



proporcionadas por Antonio Gordillo Guerrero, profesor del área de Electrónica de la Escuela Politécnica de Cáceres:

1. El módulo de gráficos calcula los estadísticos necesarios para el gráfico a partir de los datos recibidos y genera la escena.
  - **Gráfico de distribución de ángulos:** Se representan sobre una escena vacía los incrementos de las coordenadas, representados en la tabla de la figura 9 en las celdas 4, 5 y 6, además se calculan los valores máximos y mínimos de esos incrementos para situar correctamente el zoom de la visualización y calcular la longitud de los vectores que representan el origen de coordenadas, así como la malla esférica de coordenadas. Tal como se muestra en la figura 13.

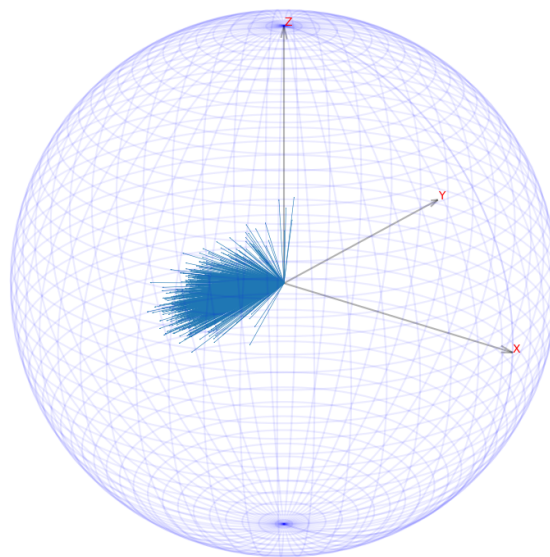


Figura 13: Ejemplo de gráfico de distribución de módulos y ángulos

- **Gráfico de densidad:** El gráfico de densidad es un mapa de calor donde se puede ver la frecuencia de las muestras en el espacio, para esto se utilizan, como en el caso anterior los valores incrementales de las muestras, igualmente se calculan los valores máximos y mínimos para enfocar correctamente la escena y dibujar una malla esférica de coordenadas. Además, en este caso, se dibujan superficies coloreadas en función de la densidad de muestras en cada zona del gráfico, para esto se ha utilizado la función de estimación de densidad del núcleo de Gauss, “kde” por sus siglas en inglés.

$$\hat{f}_h(x) = \frac{1}{n} \sum_{i=1}^n K_h(x - x_i) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right)$$

Ecuación 13: Fórmula para el cálculo del kde

Se trata de una función matemático-estadística que se define con la fórmula de la ecuación 13. Esta función está implementada en la librería *scipy* de *Python*, por lo que su implementación no ha sido necesaria.

A continuación, en la figura 14 se muestra un ejemplo del funcionamiento de este algoritmo.

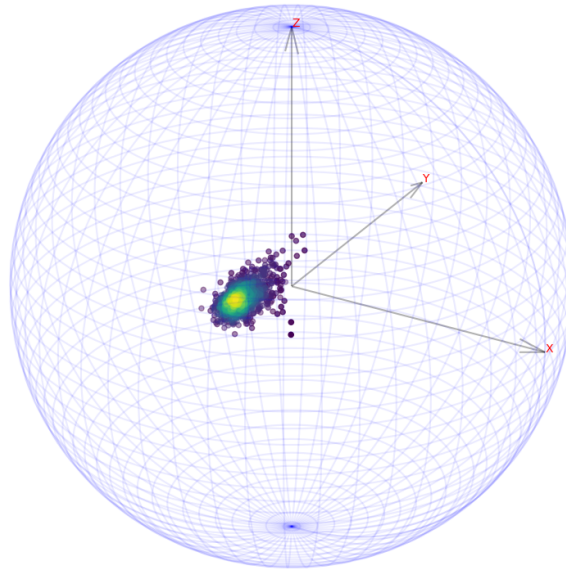


Figura 14: Ejemplo de gráfico de densidad

- **Gráfico de posición de vectores:** En este gráfico se representan los vectores resultantes del procesamiento de los datos de entrada, esto es, las últimas seis celdas de la tabla representada en la figura 9. Como en el caso anterior, este algoritmo calcula los valores máximos y mínimos para centrar la escena y aplicarle un zoom adecuado por defecto.

- Se muestra en la figura 15 un ejemplo de su funcionamiento:

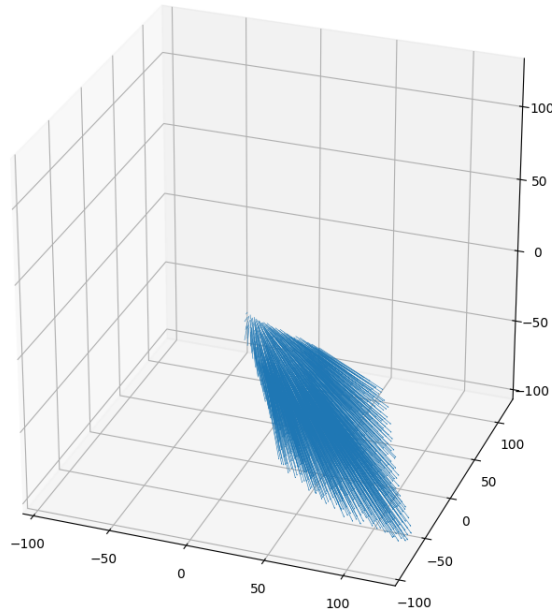


Figura 15: Ejemplo de gráfico de posición de vectores

2. Se envía a la escena los objetos representables bien formados.
3. Se muestra la escena en pantalla y se guarda la imagen en la ruta introducida por el usuario si así lo ha especificado.

### 5.3 Diseño del sistema

La librería se basa en cuatro pilares fundamentales, tres de ellos representados como clases *manager*, sitas en el paquete del mismo nombre, que se apoyan en clases de dominio inferior para ejecutar su funcionalidad y la cuarta es el paquete *graphs* que contiene el resto de *managers* responsables de la representación gráfica de los estadísticos calculados, utilizándose una clase para cada gráfico: de densidad, y distribución de módulos y ángulos, y posición de vectores.

En el paquete *mánager* encontraremos la clase *FileManager.py* responsable de la carga de datos y su transformación al formato adecuado y las clases responsables de los cálculos estadísticos, éstas divididas en dos, una encargada de los estadísticos circulares *ModuleStatisticsManager.py* y otra de los esféricos *AngleStatisticsManager.py*.

Además esta estructura está apoyada en las clases contenidas en el paquete *útil*, donde encontraremos funciones de ayuda para diferentes cálculos, representaciones gráficas y vectoriales comunes al resto de las funciones de nivel superior.

A continuación se detallan cada uno de los paquetes clases y métodos.

El proyecto está organizado en dos paquetes para facilitar su comprensión: *mánager* y *útil*.

### 5.3.1 Capa *mánager*

En la figura 16 se observan las clases públicas y de más alto nivel de la librería, que pueden ser invocadas desde cualquier sitio, a continuación se detallan sus métodos más importantes.

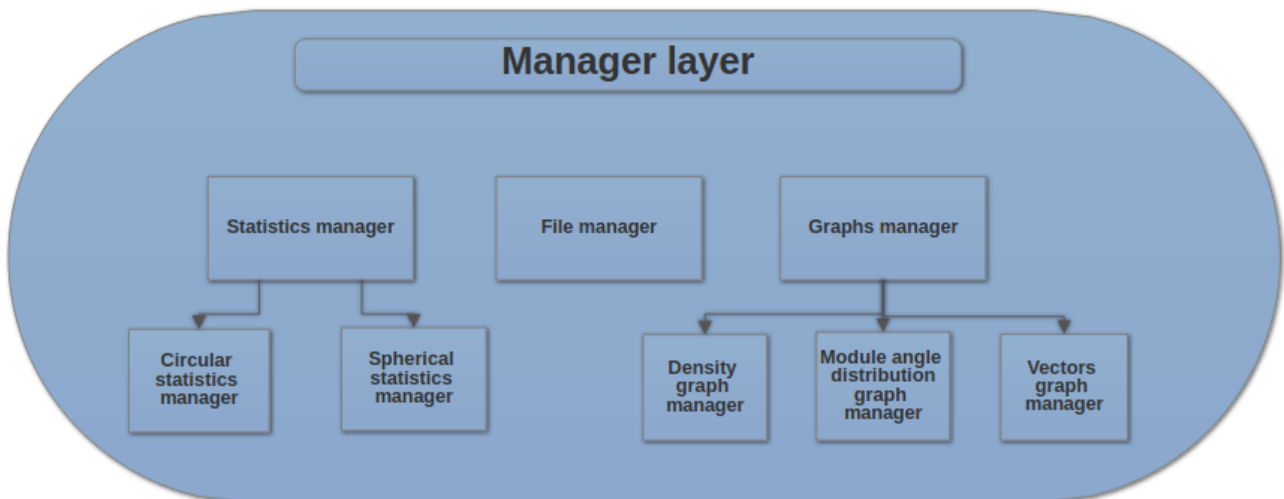


Figura 16: Diagrama de capa de *managers*

- ModuleStatisticsManager.py: El *mánager* para calcular los estadísticos lineales con los siguientes métodos:

- **module\_variance** (dat)

Calcula la varianaza del módulo según la fórmula descrita en "Ecuación 2", debe recibir por parámetros una lista compuesta por números de valores reales que represente los módulos de los vectores tridimensionales, que se pueden obtener de la primera celda de la tabla de vectores (figura 11).

- **module\_stadard\_deviation** (dat)

Calcula la desviación estándar de acuerdo con la fórmula descrita en "Ecuación 3", debe recibir por parámetros una lista compuesta por números de valores reales que represente los módulos de los vectores tridimensionales, que se pueden obtener de la primera celda de la tabla de vectores (figura 11).

- **module\_population\_variance** (dat)

Calcula la varianza poblacional descrita con la fórmula descrita en “Ecuación 4”. Debe recibir por parámetros una lista compuesta por números de valores reales que represente los módulos de los vectores tridimensionales, que se pueden obtener de la primera celda de la tabla de vectores (figura 11).

- **module\_population\_standard\_deviation** (dat)

Calcula la desviación estándar poblacional con la fórmula descrita en “Ecuación 5”. Debe recibir por parámetros una lista compuesta por números de valores reales que represente los módulos de los vectores tridimensionales, que se pueden obtener de la primera celda de la tabla de vectores (figura 11).

- **skewness\_module\_coefficient** (dat)

Calcula el coeficiente de asimetría de acuerdo con la fórmula descrita en “Ecuación 6”. Debe recibir por parámetros una lista compuesta por números de valores reales que represente los módulos de los vectores tridimensionales, que se pueden obtener de la primera celda de la tabla de vectores (figura 11).

- **kurtois\_module\_coefficient** (dat)

Calcula el coeficiente Kurtosis de acuerdo con la fórmula descrita en “Ecuación 7”. Debe recibir por parámetros una lista compuesta por números de valores reales que represente los módulos de los vectores tridimensionales, que se pueden obtener de la primera celda de la tabla de vectores (figura 11).

- [AngleStatisticsManager.py](#): El *mánager* para calcular los estadísticos esféricos con los siguientes métodos:

- **mean\_direction**(coordinates\_matrix)

Calcula la dirección media de los vectores recibidos por parámetros de acuerdo con las fórmulas descrita en “Ecuación 8”. Los vectores recibidos deben estar en forma incremental, tal como se obtienen de las posiciones 4, 5 y 6 de la tabla de vectores (figura 11).

- **mean\_module** (coordinates\_matrix)

Calcula el módulo medio de los vectores recibidos por parámetros de acuerdo a la fórmula descrita en “Ecuación 10”. Los vectores recibidos deben estar en forma incremental, tal como se obtienen de las posiciones 4, 5 y 6 de la tabla de

vectores (figura 11).

- **concentration\_parameter** (coordinates\_matrix)

Calcula el parámetro de concentración de los vectores recibidos por parámetros de acuerdo a la fórmula descrita en “Ecuación 11”.

Los vectores recibidos deben estar en forma incremental, tal como se obtienen de las posiciones 4, 5 y 6 de la tabla de vectores (figura 11).

- **spherical\_error** (coordinates\_matrix)

Calcula el valor del error esférico de los vectores recibidos por parámetros de acuerdo a la fórmula descrita en “Ecuación 12”.

Los vectores recibidos deben estar en forma incremental, tal como se obtienen de las posiciones 4, 5 y 6 de la tabla de vectores (figura 11).

- FileManager.py: El *mánager* para la gestión de ficheros de texto, sus métodos más importantes son los siguientes:

- **read\_file** ()

Pide al usuario la ruta del fichero de datos de entrada, los lee y los almacena en una estructura de datos.

- **load\_data** (vector\_matrix)

Procesa los datos que se han leído previamente, y debe recibir por parámetros, transformándolos al formato que acepta la aplicación y sirviéndolo en la estructura de datos comentada en “Figura 11”.

- DensityGraphManager.py: El *mánager* que se encarga del dibujado del mapa de densidad, sólo contiene este método:

- **draw\_density\_graph** (dat)

Debe recibir por parámetros la estructura de datos devuelta por el método “load\_data”. Genera la escena del mapa de densidad y guarda una imagen del gráfico en alta resolución si el usuario lo solicita.

- ModuleAngleGraphManager.py: El *mánager* que se encarga del dibujado del gráfico de distribución de módulos y ángulos, sólo contiene este método:

- **draw\_module\_angle\_distrib** (dat)

Debe recibir por parámetros la estructura de datos devuelta por

el método "load\_data". Genera la escena del mapa de distribución de ángulos y módulos y guarda una imagen del gráfico en alta resolución si el usuario lo solicita.

- VectorGraph.py: El mánager que se encarga del dibujado del gráfico de posición de vectores, sólo contiene este método:

- **draw\_vector\_graph** (dat)

Debe recibir por parámetros la estructura de datos devuelta por el método "load\_data". Genera la escena del mapa de posición de vectores y guarda una imagen del gráfico en alta resolución si el usuario lo solicita.

### 5.3.2 Capa útil

En la figura 17 se observan las tres clases pertenecientes a la capa útil, que son las responsables de dar soporte a las funciones de más alto nivel, aquí se encuentran métodos de ayuda clasificados en los campos aritmético, vectorial y de dibujo que se explican a continuación.

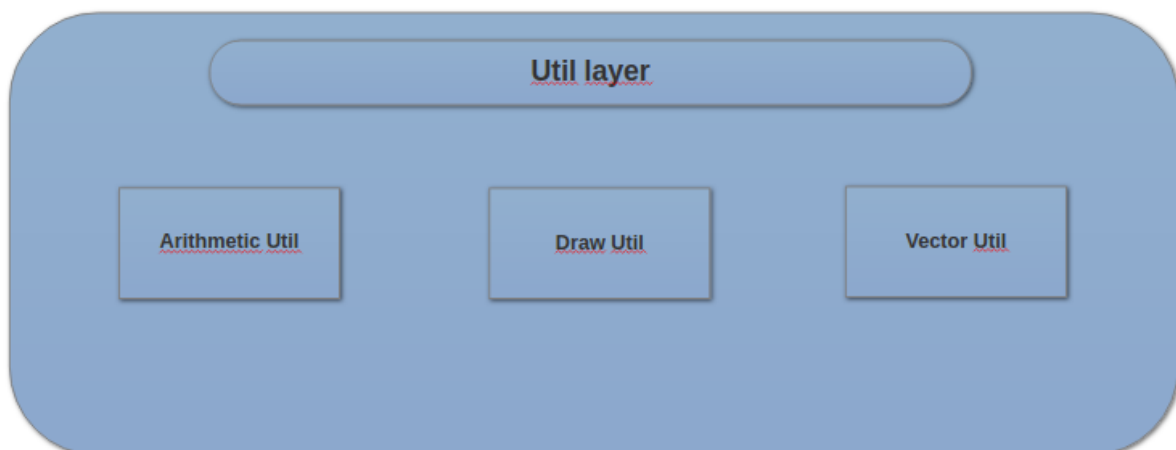


Figura 17: Diagrama de capa útil

- ArithmeticUtil.py: En esta clase se encuentran las funciones de ayuda aritmética, como la transformación de grados a radianes, cálculo de media aritmética, valores máximos y mínimos. Todas estas funciones se apoyan en la librería Numpy.
- VectorUtil.py: Contiene las funciones de soporte para las operaciones con vectores: cálculo del elemento mayor, transformación a valores polares, y cálculo de módulos.
- DrawUtil.py: Donde se encuentran las funciones de soporte para el dibujado

de gráficos: el cálculo del origen de coordenadas, el dibujo de la matriz de coordenadas esférica y el dibujo de los vectores del eje.



## Capítulo 6: Resultados

Mediante el uso de esta librería se han analizado varias estaciones pertenecientes a la red EUREF, que se encuentran bajo territorio español, se muestran en la tabla 1, que describe el nombre de la estación, la localidad donde se encuentra así como su fecha de primera y última emisión. Al estar todas las estaciones operativas, la fecha de última emisión es siempre coincidente con la fecha de finalización de este documento: Noviembre de 2018, y la de primera emisión es el momento en el que se montó la estación, aquí sí se pueden observar grandes variaciones entre territorios dentro del Estado, pudiendo acceder a estaciones con un gran histórico de datos, desde 1999 y otras, como por ejemplo la de Ibiza, que se estableció en 2017 de la que apenas encontramos datos históricos, debido a lo cual la hemos descartado del caso de estudio por no aportar valores relevantes.

Debido a la gran cantidad de datos, se ha decidido tomar una muestra por mes de ocho de estas estaciones, cuyo aspecto se observa en la figura 18, que se encuentran distribuidas por los márgenes del territorio de la península ibérica, los resultados obtenidos se muestran en forma de tabla, tal como se desarrolla, en las siguientes páginas.



Figura 18: Estación GNSS de Cáceres en la Escuela Politécnica

Nombre de la estación	Localidad	Primera emisión	Última emisión
ACOR	A Coruña	Abril 1999	Noviembre 2018
ALAC	Alicante	Marzo 1999	Noviembre 2018
ALME	Almería	Diciembre 2000	Noviembre 2018
ALBA	Albacete	Abril 2007	Noviembre 2018
BCLN	Barcelona	Abril 2015	Noviembre 2018
CACE	Cáceres	Marzo 2001	Noviembre 2018
CANT	Santander	Diciembre 2000	Noviembre 2018
CEU1	Ceuta	Febrero 2008	Noviembre 2018
COBA	Córdoba	Febrero 2005	Noviembre 2018
HUEL	Huelva	Marzo 2005	Noviembre 2018
IBIZ	Eivessa	Febrero 2017	Noviembre 2018
IZAN	Guímar	Marzo 2008	Noviembre 2018
LEON	León	Marzo 2007	Noviembre 2018
LPA1	Garafía	Junio 2001	Noviembre 2018
MALA	Málaga	Septiembre 2005	Noviembre 2018
MELI	Melilla	Mayo 2011	Noviembre 2018
RIO1	Logroño	Mayo 2012	Noviembre 2018
SALA	Salamanca	Septiembre 2006	Noviembre 2018
SONS	Sonseca	Abril 2007	Noviembre 2018
TARI	Tarifa	Julio 2015	Noviembre 2018
TERU	Teruel	Abril 2008	Noviembre 2018
VALA	Valladolid	Febrero 2008	Noviembre 2018
VALE	Valencia	Diciembre 2000	Noviembre 2018
VIGO	Vigo	Abril 2005	Noviembre 2018
YEBE	Yebes	Septiembre 2000	Noviembre 2018
ZARA	Zaragoza	Julio 2006	Noviembre 2018

*Tabla 1: Estaciones EUREF en España*

En este análisis se puede observar una clara tendencia positiva en el eje Y y Z, y una tendencia negativa en el eje X puesto que las correspondencias son:

- Eje y: Norte-Sur
- Eje x: Este-Oeste
- Eje z: Altura

Se podría deducir que la tendencia del movimiento de la placa tectónica en la que se encuentra situada la Península Ibérica es Norte-Oeste y además tiende a ganar altura con el paso del tiempo, aunque esto es un análisis superficial que debería ser analizado en más detalle por expertos en geodesia.

Por otra parte, aunque *a priori* se pudiera esperar que los datos variaran poco de una estación a otra, se observa que la realidad es distinta, deducimos que esta varianza en los resultados se debe al terreno en que se encuentran situados los monumentos (estaciones GNSS). Por ejemplo, la estación de Cáceres está montada en la terraza de uno de los pabellones de la Escuela Politécnica, que se encuentra sobre terreno arcilloso, es por tanto, presumible, que esta estación presente unas variaciones distintas a otra que esté asentada en un terreno mas firme.

Se puede ver, en la cabecera de la tabla, la localidad donde se encuentra posicionada la estación, el nombre en la red EUREF, la fecha de primera emisión y la reseña de la estación obtenida del IGN, como se aprecia en la figura 19.

Posición	Nombre	Primera emisión	Última emisión	Reseña de la estación												
Cáceres	<b>CACE</b>	Marzo 2001	Noviembre 2018	<table border="1"> <tr> <td>Longitud.....: - 6° 20' 30,42692"</td> <td>X.....: 4899866.803 m.</td> </tr> <tr> <td>Latitud.....: 39° 28' 43,89817"</td> <td>Y.....: -544567.603 m.</td> </tr> <tr> <td>Altitud elipsoidal: 436.501 m.</td> <td>Z.....: 4033769.767 m.</td> </tr> <tr> <td>X UTM.....: 728635.783 m.</td> <td>Altitud sobre el nivel medio del mar:</td> </tr> <tr> <td>Y UTM.....: 4373291.864 m.</td> <td>382.365 m.</td> </tr> <tr> <td>Huso.....: 29</td> <td></td> </tr> </table>	Longitud.....: - 6° 20' 30,42692"	X.....: 4899866.803 m.	Latitud.....: 39° 28' 43,89817"	Y.....: -544567.603 m.	Altitud elipsoidal: 436.501 m.	Z.....: 4033769.767 m.	X UTM.....: 728635.783 m.	Altitud sobre el nivel medio del mar:	Y UTM.....: 4373291.864 m.	382.365 m.	Huso.....: 29	
Longitud.....: - 6° 20' 30,42692"	X.....: 4899866.803 m.															
Latitud.....: 39° 28' 43,89817"	Y.....: -544567.603 m.															
Altitud elipsoidal: 436.501 m.	Z.....: 4033769.767 m.															
X UTM.....: 728635.783 m.	Altitud sobre el nivel medio del mar:															
Y UTM.....: 4373291.864 m.	382.365 m.															
Huso.....: 29																

Figura 19: Cabecera de tabla de resultados de la estación de Cáceres

En el resto de filas de la tabla encontramos, en la parte izquierda, los gráficos obtenidos por medio de la librería, de distribución de ángulos y de densidad, y en la columna derecha el análisis numérico, tanto modular como angular. Así como se muestra en la figura 20.

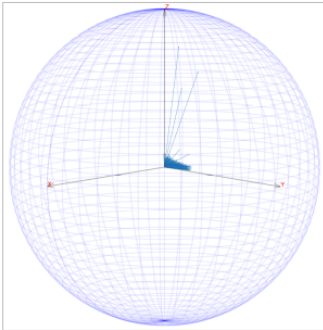
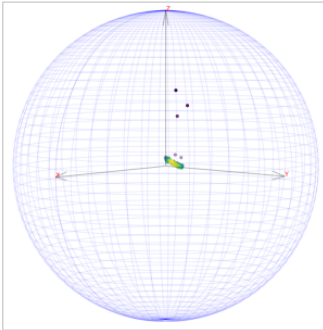
Análisis gráfico	Análisis numérico
Distribución de módulos y ángulos	Modulares
	Número de elementos = 201 Valor mínimo = 0.00310 Valor máximo = 0.06003 Rango = 0.05693 Media aritmética = 0.00792 Error medio estándar = 0.00040 Desviación estándar = 0.00580 Varianza = $3.37495 \times 10^{-5}$ Desv. estándar poblacional = 0.00580 Varianza poblacional = $3.35816 \times 10^{-5}$ Coeficiente Skewness = 6.57006 Coeficiente Kurtosis = 51.00158
Gráfico de densidad	Angulares
	<b>Vectores reales</b> Colatitud = 55.06607 Longitud = 107.30293 Media modular = 0.01265  <b>Vectores unitarios</b> Colatitud = 36.45905 Longitud = 290.41633 Media modular = 0.90280 Param. de concentración = 10.20819 Error estándar esférico = 3.48921

Figura 20: Cuerpo de tabla de resultados de la estación de Cáceres

Además, se incluye antes del pie de tabla, la variación de la posición a lo largo del periodo de tiempo representada bidimensionalmente, siempre obtenida de la página web de la agencia europea EUREF [9] como se muestra en la figura 21.

### Serie de posición temporal

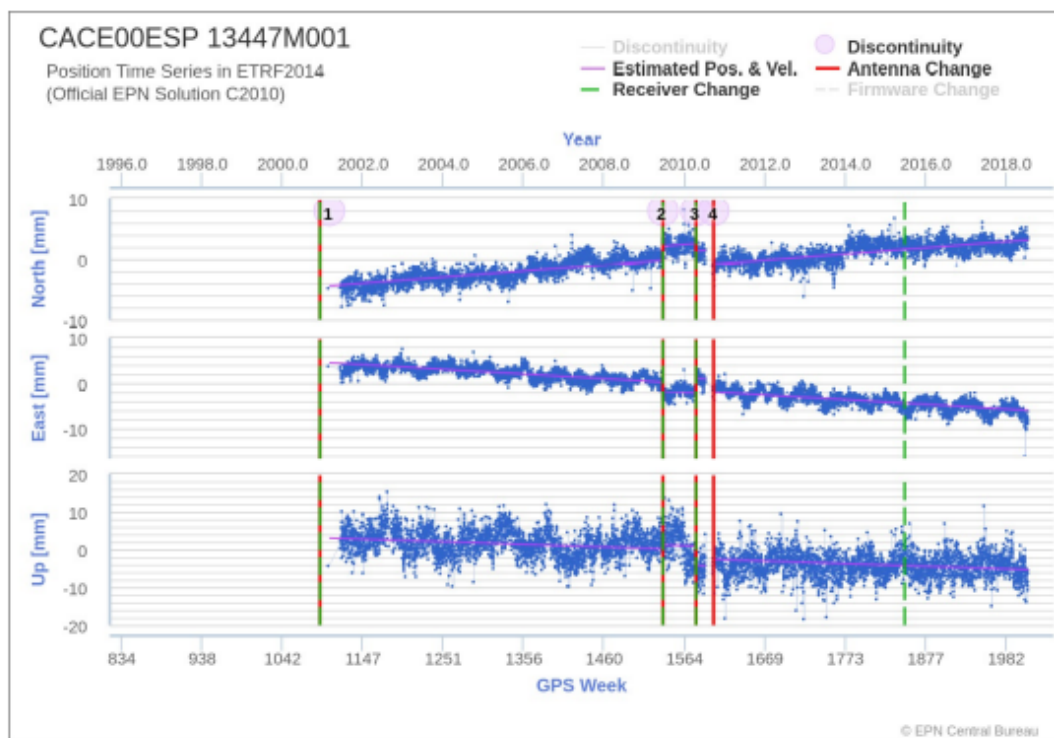
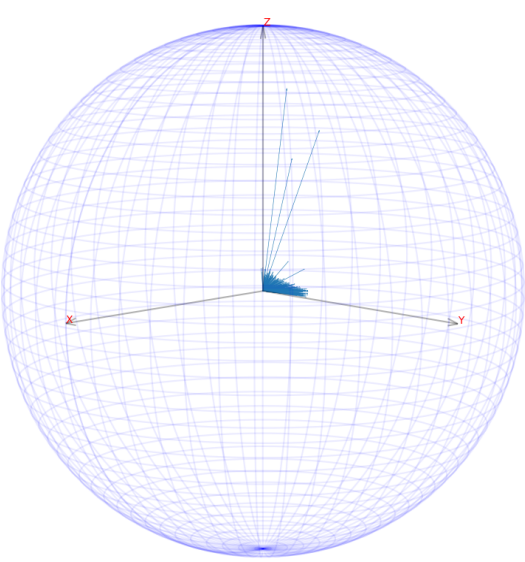
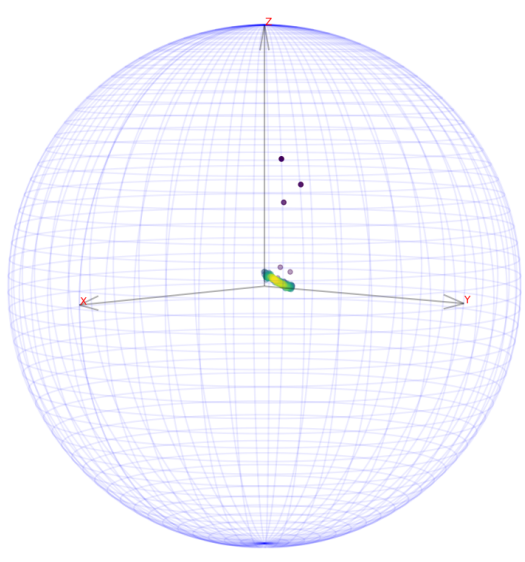


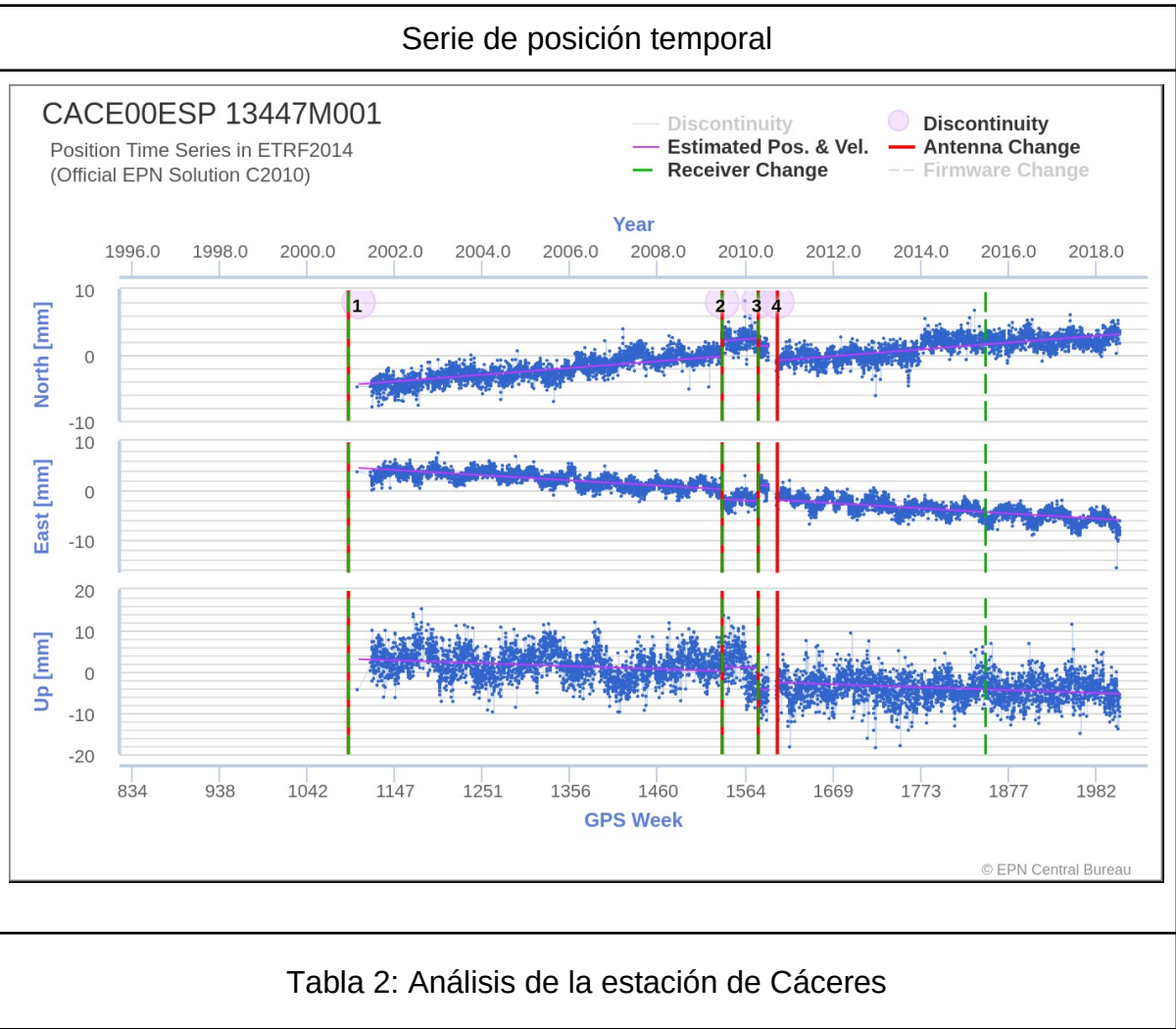
Figura 21: Gráfica bidimensional de posición temporal de la estación de Cáceres

En la figura 21 también se observan líneas verticales que coinciden con referencias numéricas que aparecerán en las siguientes tablas. Su significado es, para la primera línea (número 1) el momento de comienzo de emisión de la estación, el resto son momentos de interrupción del servicio, cambios de antena, actualizaciones del firmware... identificados con una leyenda de color como está descrito en la propia imagen. En el eje horizontal superior representa el año desde 1996 y el inferior la fecha en formato “Semana GPS” del que se habla con más detalle en el Anexo II.

La tabla 2, referente a la estación de Cáceres, se observa una tendencia positiva en el eje Y-Z y negativa en X. En el gráfico de densidad podemos observar algunos *outliers*, tomando valores muy altos en Z.

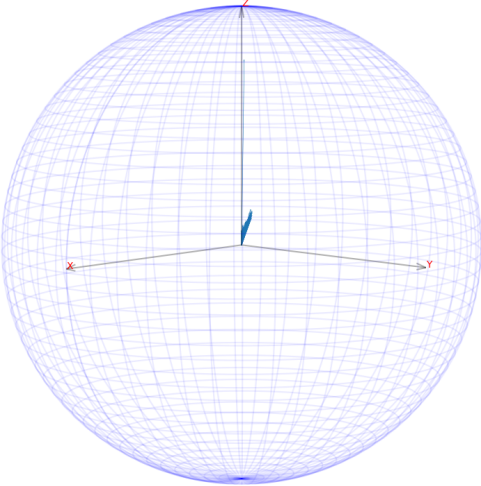
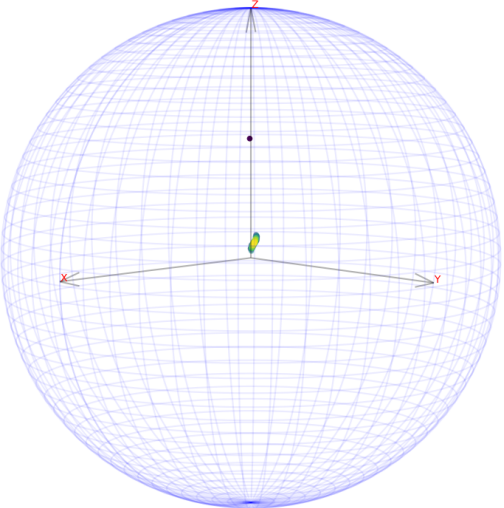
Posición	Nombre	Primera emisión	Última emisión	Reseña de la estación
Cáceres	<b>CACE</b>	Marzo 2001	Noviembre 2018	<p>Longitud.....: - 6° 20' 30,42692" X.....: 4899866.803 m.            Latitud.....: 39° 28' 43,89817" Y.....: -544567.603 m.            Altitud elipsoidal: 436.501 m. Z.....: 4033769.767 m.</p> <hr/> <p>X UTM.....: 728635.783 m. Altitud sobre el nivel medio del mar:            Y UTM.....: 4373291.864 m.            Huso.....: 29 382.365 m.</p>
Análisis gráfico				Análisis numérico
Distribución de módulos y ángulos				Modulares
				<p>Número de elementos = 201            Valor mínimo = 0.00310            Valor máximo = 0.06003            Rango = 0.05693            Media aritmética = 0.00792            Error medio estándar = 0.00040            Desviación estándar = 0.00580            Varianza = 3.37495 x10<sup>-5</sup>            Desv. estándar poblacional = 0.00580            Varianza poblacional = 3.35816 x10<sup>-5</sup>            Coeficiente Skewness = 6.57006            Coeficiente Kurtosis = 51.00158</p>
Gráfico de densidad				Angulares
				<p><b>Vectores reales</b></p> <p>Colatitud = 55.06607            Longitud = 107.30293            Media modular = 0.01265</p> <p><b>Vectores unitarios</b></p> <p>Colatitud = 36.45905            Longitud = 290.41633            Media modular = 0.90280            Param. de concentración = 10.20819            Error estándar esférico = 3.48921</p>

## Serie de posición temporal



Además se observan también tres interrupciones de servicio temporal entre los años 2009 y 2011.

La tabla 3, referente a la estación de Sonseca, se observa una tendencia positiva en el eje Y-Z y negativa en X. En el gráfico de densidad podemos observar gran cantidad de valores cercanos al centro de coordenadas con un *outlier* tomando valores muy altos en Z.

Posición	Nombre	Primera emisión	Última emisión	Reseña de la estación
Sonseca	<b>SONS</b>	Abril 2007	Noviembre 2018	Longitud.....: - 3° 57' 50,29100" Latitud.....: 39° 40' 31,26349" Altitud elipsoidal: 808.944 m. X.....: 4904660.506 m. Y.....: -339868.046 m. Z.....: 4050823.489 m. X UTM.....: 417326.754 m. Y UTM.....: 4392169.397 m. Huso.....: 30 Altitud sobre el nivel medio del mar:
Análisis gráfico				Análisis numérico
Distribución de módulos y ángulos				Modulares
				Número de elementos = 128 Valor mínimo = 0.00367 Valor máximo = 0.06229 Rango = 0.05861 Media aritmética = 0.00883 Error medio estándar = 0.00046 Desviación estándar = 0.00529 Varianza = $2.80602 \times 10^{-5}$ Desv. estándar poblacional = 0.00529 Varianza poblacional = $2.78410 \times 10^{-5}$ Coeficiente Skewness = 8.18257 Coeficiente Kurtosis = 82.59939
Gráfico de densidad				Angulares
				<p><b>Vectores reales</b></p> Colatitud = 29.21153 Longitud = 64.82413 Media modular = 0.00880
				<p><b>Vectores unitarios</b></p> Colatitud = 41.37542 Longitud = 63.84010 Media modular = 0.99633 Param. de concentración = 270.90811 Error estándar esférico = 0.75004



### Serie de posición temporal

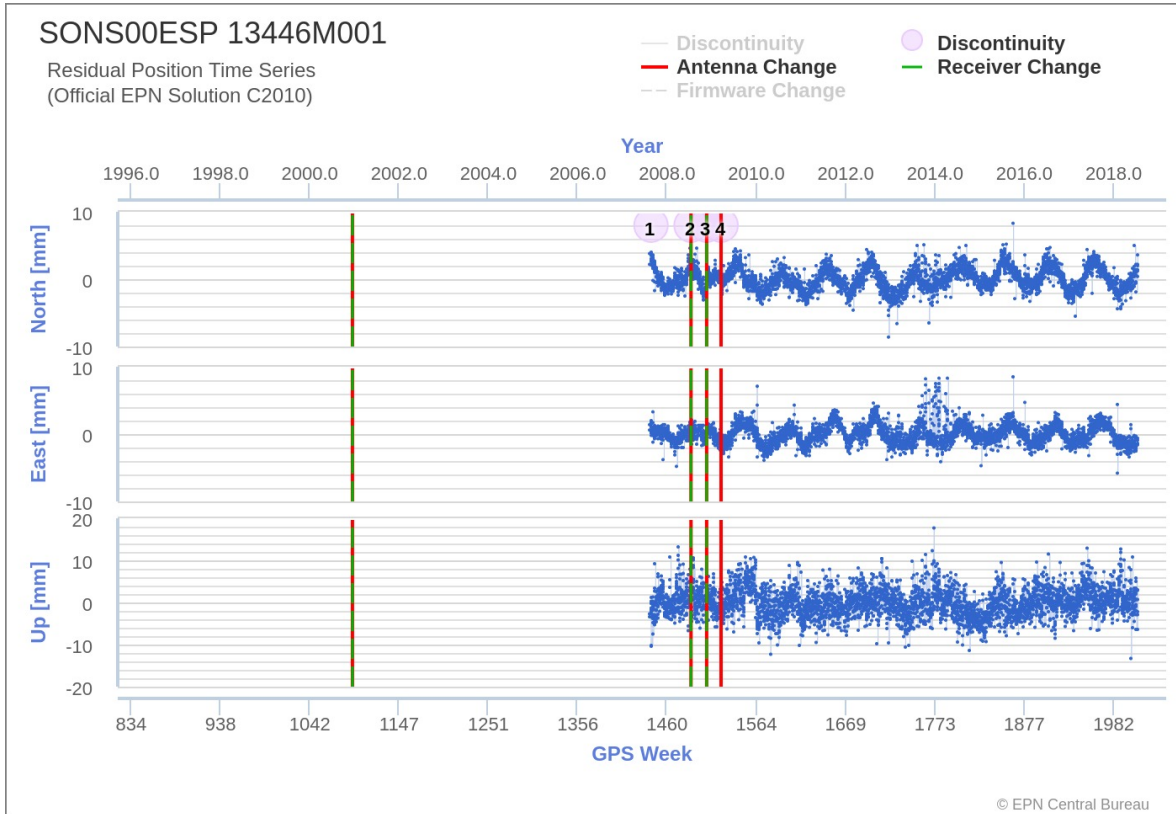
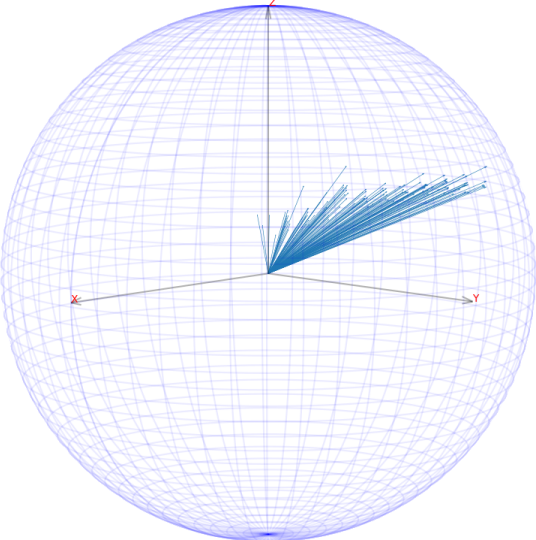
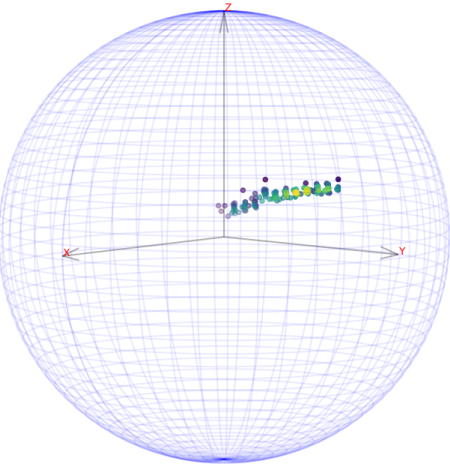


Tabla 3: Análisis de la estación de Sonseca

Se aprecia también un periodo de alta discontinuidad del servicio entre los años 2008 y 2010.

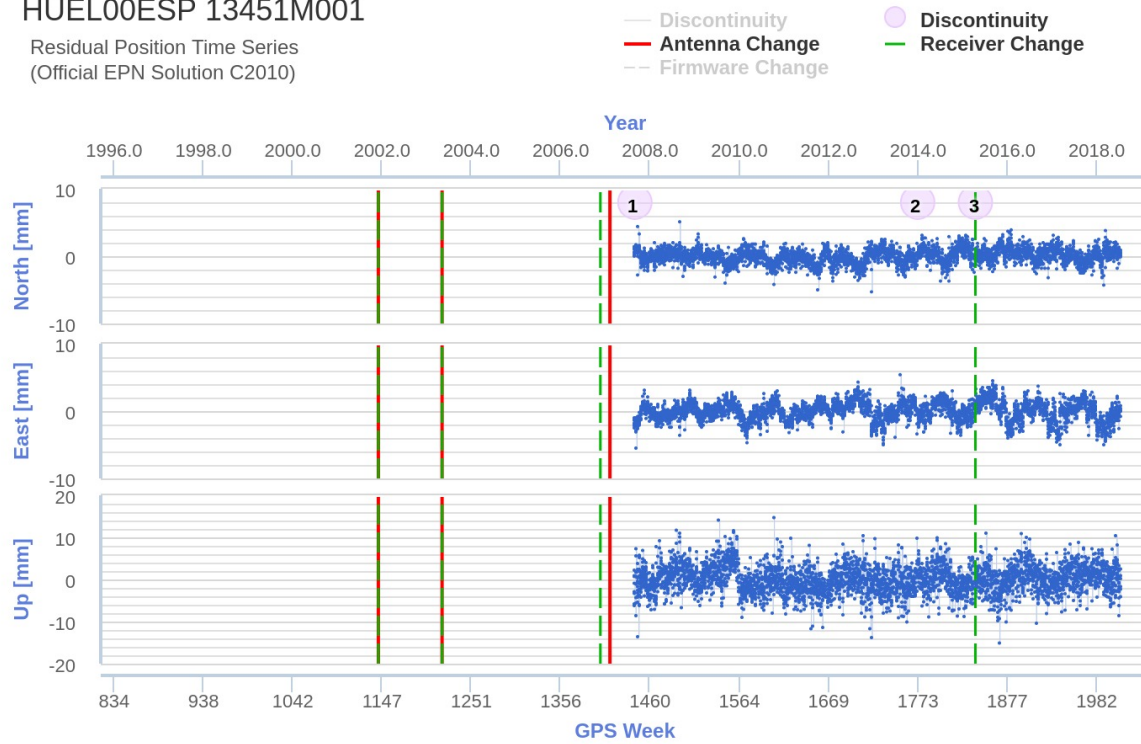
La tabla 4, referente a la estación de Huelva, se observa una tendencia positiva en el eje Y-Z muy superior a la de las tablas anteriores, y negativa en X. En el gráfico de densidad observamos valores más dispersos que en las anteriores.

Posición	Nombre	Primera emisión	Última emisión	Reseña de la estación
Huelva	<b>HUEL</b>	Marzo 2005	Noviembre 2018	<p>Longitud.....: - 2° 27' 34,02005" X.....: 5105220.286 m.            Latitud.....: 36° 51' 09,11227" Y.....: -219278.834 m.            Altitud elipsoidal: 127.499 m. Z.....: 3804386.869 m.</p> <p>X UTM.....: 548188.938 m. Altitud sobre el nivel medio del mar:            Y UTM.....: 4078649.716 m.            Huso.....: 30 77.702 m.</p>
Análisis gráfico				Análisis numérico
Distribución de módulos y ángulos				Modulares
				<p>Número de elementos = 201            Valor mínimo = 0.00310            Valor máximo = 0.06003            Rango = 0.05693            Media aritmética = 0.00792            Error medio estándar = 0.00040            Desviación estándar = 0.00580            Varianza = 3.37495 x10<sup>-5</sup>            Desv. estándar poblacional = 0.00580            Varianza poblacional = 3.35816 x10<sup>-5</sup>            Coeficiente Skewness = 6.57006            Coeficiente Kurtosis = 51.00158</p>
Gráfico de densidad				Angulares
				<p><b>Vectores reales</b></p> <p>Colatitud = 55.06607            Longitud = 107.30293            Media modular = 0.01265</p> <p><b>Vectores unitarios</b></p> <p>Colatitud = 36.45905            Longitud = 290.41633            Media modular = 0.90280            Param. de concentración = 10.20819            Error estándar esférico = 3.48921</p>

## Serie de posición temporal

HUEL00ESP 13451M001

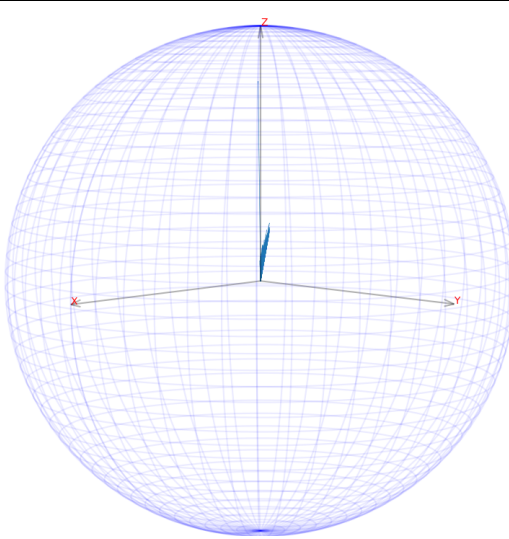
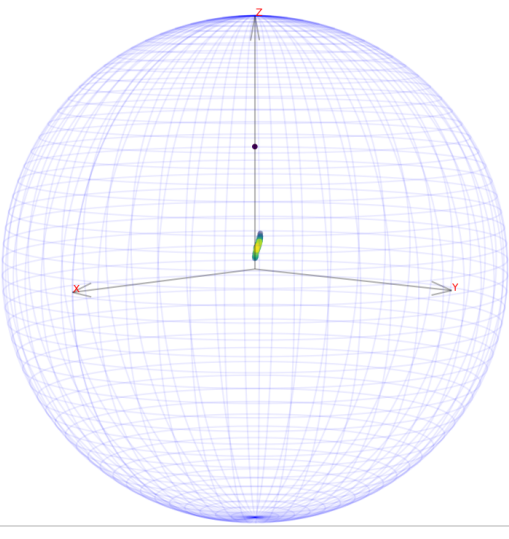
Residual Position Time Series  
(Official EPN Solution C2010)



© EPN Central Bureau

Tabla 4: Análisis de la estación de Huelva

La tabla 5, referente a la estación de Vigo, se observa una tendencia positiva en el eje Y-Z y negativa en X. En el gráfico de densidad se observa un *outlier* con valor alto en el eje Z.

Posición	Nombre	Primera emisión	Última emisión	Reseña de la estación
Vigo	<b>VIGO</b>	Abril 2005	Noviembre 2018	<p>Longitud.....: - 8° 48' 47,05821"   X.....: 4677481.058 m.</p> <p>Latitud.....: 42° 11' 02,31813"   Y.....: -725205.081 m.</p> <p>Altitud elipsoidal: 87.761 m.   Z.....: 4260827.167 m.</p> <hr/> <p>X UTM.....: 515436.285 m.   Altitud sobre el nivel medio del mar:</p> <p>Y UTM.....: 4670220.247 m.</p> <p>Huso.....: 29</p>
Análisis gráfico				Análisis numérico
Distribución de módulos y ángulos				Modulares
				<p>Número de elementos = 149</p> <p>Valor mínimo = 0.00522</p> <p>Valor máximo = 0.05960</p> <p>Rango = 0.05437</p> <p>Media aritmética = 0.01056</p> <p>Error medio estándar = 0.00038</p> <p>Desviación estándar = 0.00474</p> <p>Varianza = 2.25196 x10<sup>-5</sup></p> <p>Desv. estándar poblacional = 0.00474</p> <p>Varianza poblacional = 2.23684 x10<sup>-5</sup></p> <p>Coficiente Skewness = 7.50193</p> <p>Coficiente Kurtosis = 77.36163</p>
Gráfico de densidad				Angulares
				<p><b>Vectores reales</b></p> <p>Colatitud = 7.37575</p> <p>Longitud = 105.25682</p> <p>Media modular = 0.01046</p> <p><b>Vectores unitarios</b></p> <p>Colatitud = 21.38909</p> <p>Longitud = 358.90951</p> <p>Media modular = 0.86307</p> <p>Param. de concentración = 7.25423</p> <p>Error estándar esférico = 4.66937</p>

## Serie de posición temporal

VIGO00ESP 13450M001

Residual Position Time Series  
(Official EPN Solution C2010)

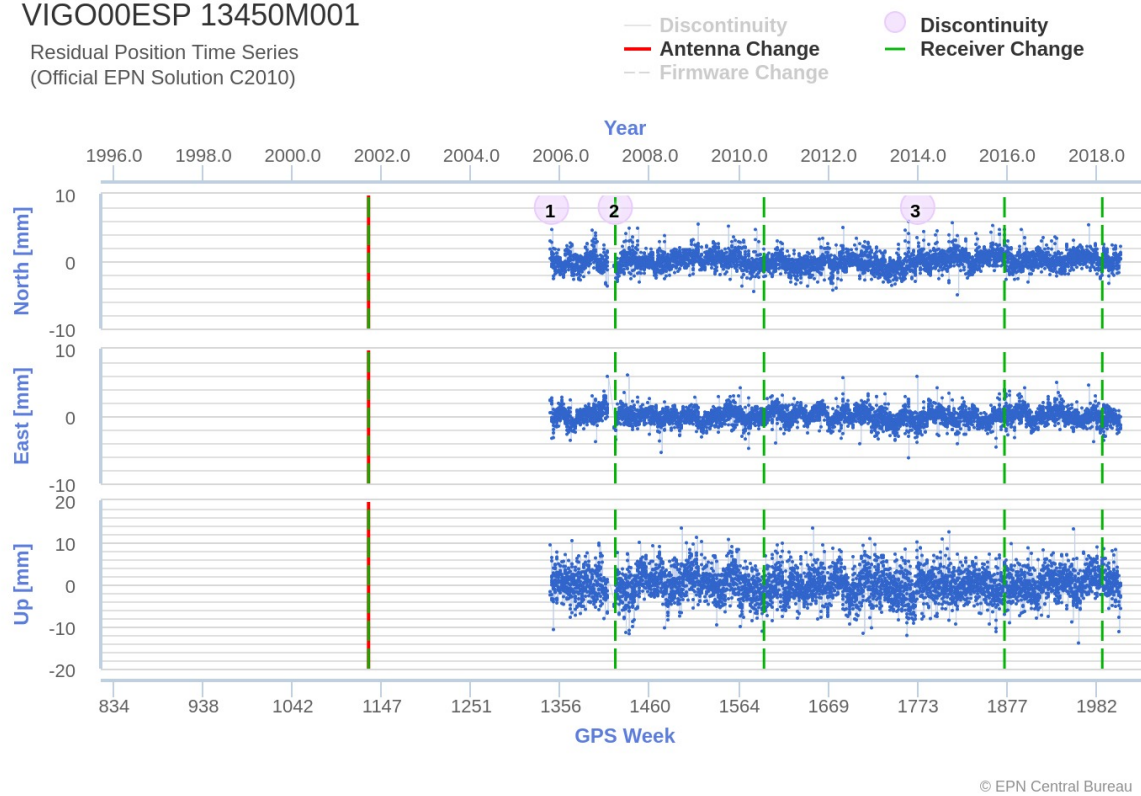
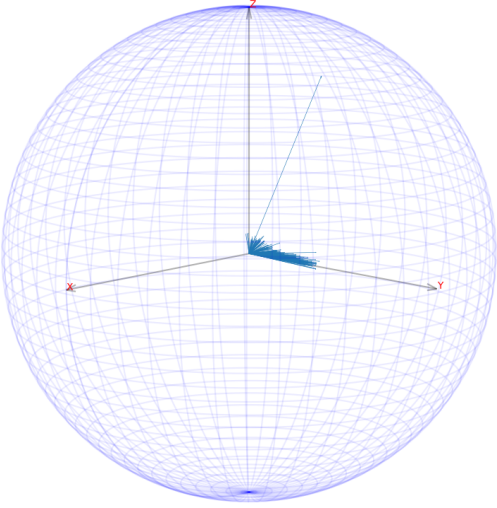
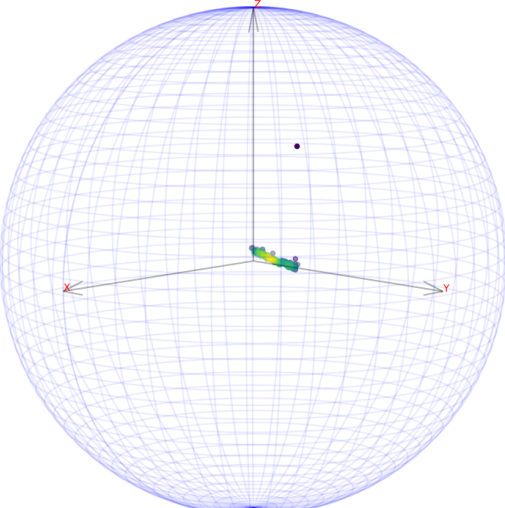


Tabla 5: Análisis de la estación de Vigo

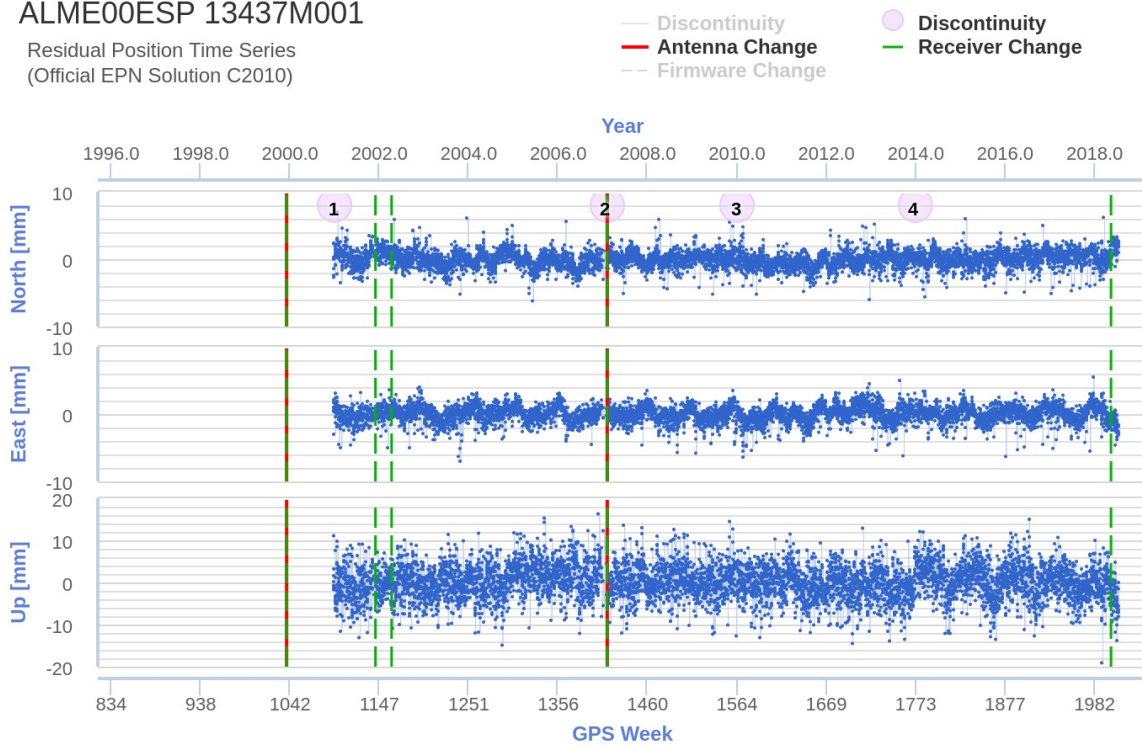
La tabla 6, referente a la estación de Almería, se observa una tendencia positiva en el eje Y-Z y negativa en X. En el gráfico de densidad se observa un *outlier* con valor alto en el eje Z.

Posición	Nombre	Primera emisión	Última emisión	Reseña de la estación
Almería	<b>ALME</b>	Diciembre 2000	Noviembre 2018	<p> <b>Longitud</b>.....: - 2° 27' 34,02005"    <b>X</b>.....: 5105220.286 m.  <b>Latitud</b>.....: 36° 51' 09,11227"    <b>Y</b>.....: -219278.834 m.  <b>Altitud elipsoidal</b>: 127.499 m.    <b>Z</b>.....: 3804386.869 m. </p> <hr/> <p> <b>X UTM</b>.....: 548188.938 m.    <b>Altitud sobre el nivel medio del mar</b>:  <b>Y UTM</b>.....: 4078649.716 m.    <b>77.702 m.</b>  <b>Huso</b>.....: 30 </p>
<b>Análisis gráfico</b>				<b>Análisis numérico</b>
<b>Distribución de módulos y ángulos</b>				<b>Modulares</b>
				<p> Número de elementos = 208  Valor mínimo = 0.00419  Valor máximo = 0.07863  Rango = 0.07443  Media aritmética = 0.01534  Error medio estándar = 0.00055  Desviación estándar = 0.00798  Varianza = 6.37638 x10<sup>-5</sup>  Desv. estándar poblacional = 0.00798  Varianza poblacional = 6.34573 x10<sup>-5</sup>  Coeficiente Skewness = 2.48801  Coeficiente Kurtosis = 17.44900 </p>
<b>Gráfico de densidad</b>				<b>Angulares</b>
				<p style="text-align: center;"><b>Vectores reales</b></p> <p> Colatitud = 79.93229  Longitud = 91.03352  Media modular = 0.01465 </p> <p style="text-align: center;"><b>Vectores unitarios</b></p> <p> Colatitud = 17.31439  Longitud = 289.83955  Media modular = 0.74877  Param. de concentración = 3.96141  Error estándar esférico = 5.88021 </p>

## Serie de posición temporal

ALME00ESP 13437M001

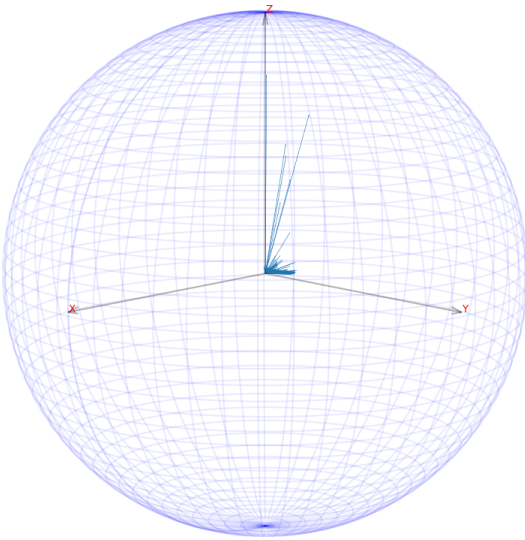
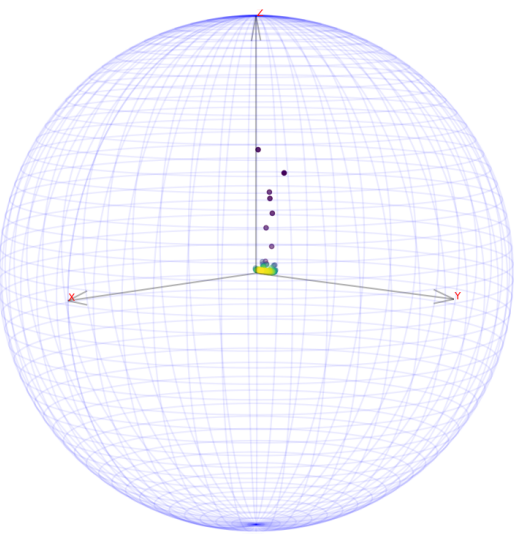
Residual Position Time Series  
(Official EPN Solution C2010)



© EPN Central Bureau

Tabla 6: Análisis de la estación de Almería

La tabla 7, referente a la estación de Valencia, se observa una tendencia positiva en el eje Y-Z y negativa en X. En el gráfico de densidad se observan varios *outliers*, algunos de ellos con valor alto en el eje Z.

Posición	Nombre	Primera emisión	Última emisión	Reseña de la estación
Valencia	<b>VALE</b>	Diciembre 2000	Noviembre 2018	<p>Longitud.....: - 0° 20' 15,54418"    X.....: 4929534.024 m.            Latitud.....: 39° 28' 50,97025"    Y.....: -29050.694 m.            Altitud elipsoidal: 77.559 m.    Z.....: 4033709.900 m.</p> <hr/> <p>X UTM.....: 728984.964 m.    Altitud sobre el nivel medio del mar:            Y UTM.....: 4373520.432 m.    mar:            Huso.....: 30    27.625 m.</p>
Análisis gráfico				Análisis numérico
Distribución de módulos y ángulos				Modulares
				<p>Número de elementos = 200            Valor mínimo = 0.00299            Valor máximo = 0.13440            Rango = 0.13140            Media aritmética = 0.01638            Error medio estándar = 0.00116            Desviación estándar = 0.01643            Varianza = 0.00027            Desv. estándar poblacional = 0.01643            Varianza poblacional = 0.00026            Coeficiente Skewness = 4.71908            Coeficiente Kurtosis = 27.23428</p>
Gráfico de densidad				Angulares
				<p><b>Vectores reales</b></p> <p>Colatitud = 64.33774            Longitud = 85.34227            Media modular = 0.01519</p> <p><b>Vectores unitarios</b></p> <p>Colatitud = 19.62950            Longitud = 73.43005            Media modular = 0.76000            Param. de concentración = 4.14587            Error estándar esférico = 5.66748</p>



## Serie de posición temporal

VALE00ESP 13439M001

Residual Position Time Series  
(Official EPN Solution C2010)

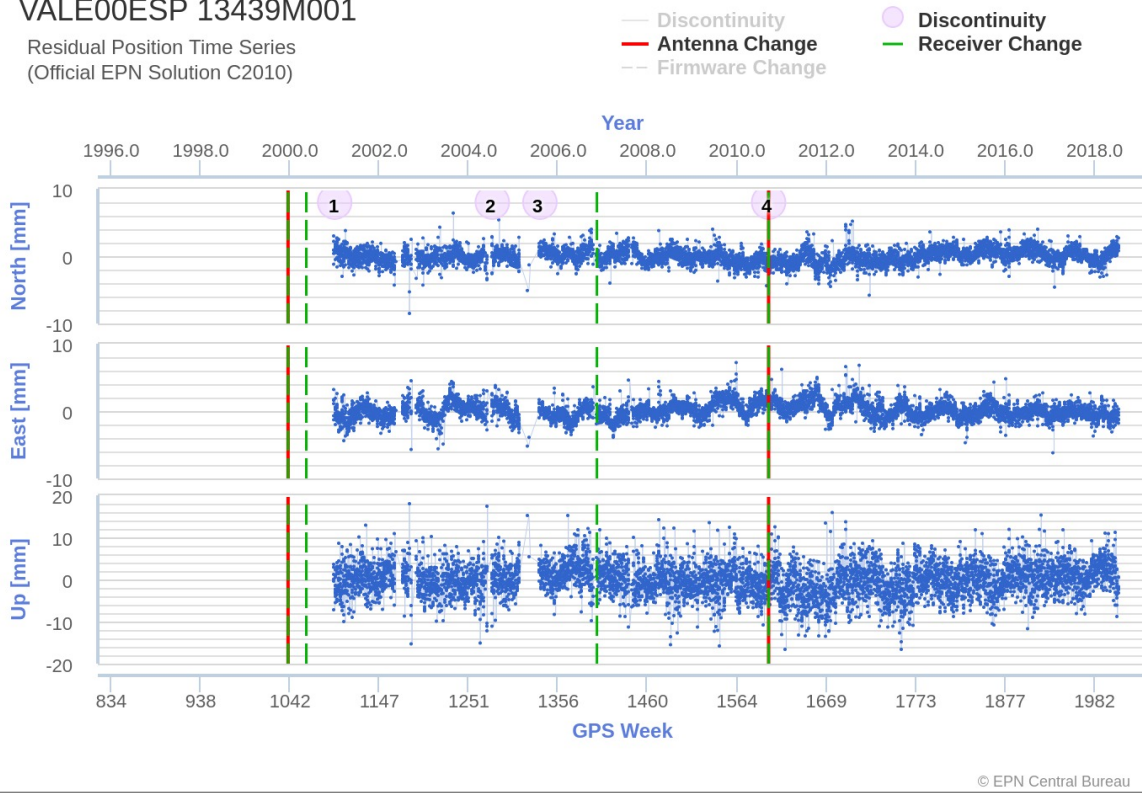
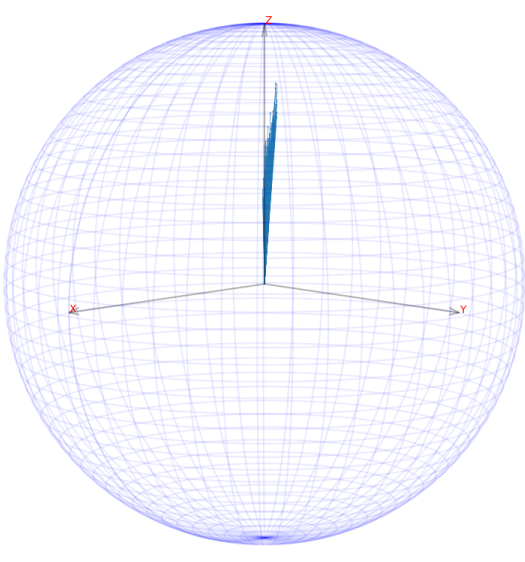
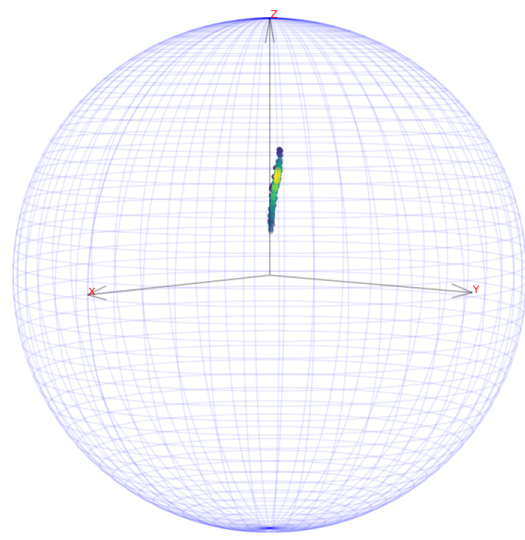


Tabla 7: Análisis de la estación de Valencia

Observamos también una discontinuidad del servicio en torno al año 2005.

La tabla 8, referente a la estación de Almería, se observa una tendencia positiva en el eje Y-Z, tomando valores muy altos para Z, y negativa en X. En el gráfico de densidad se aprecia una distribución bastante continua.

Posición	Nombre	Primera emisión	Última emisión	Reseña de la estación
Zaragoza	<b>ZARA</b>	Julio 2000	Noviembre 2018	<p>Longitud.....: - 0° 52' 55,79334" X.....: 4773803.537 m.            Latitud.....: 41° 38' 00,22164" Y.....: -73506.537 m.            Altitud elipsoidal: 296.094 m. Z.....: 4215453.678 m.</p> <p>X UTM.....: 676401.977 m. Altitud sobre el nivel medio del mar:            Y UTM.....: 4611240.257 m. mar:            Huso.....: 30 246.207 m.</p>
Análisis gráfico				Análisis numérico
Distribución de módulos y ángulos				Modulares
 <p>Un gráfico de densidad en un sistema de coordenadas 3D (X, Y, Z). El eje Z apunta hacia arriba, Y hacia la derecha y X hacia la izquierda. Se muestra una distribución de puntos que forma una columna vertical estrecha y alta, ubicada cerca del eje Z positivo, indicando una fuerte tendencia positiva en el eje Z.</p>				<p>Número de elementos = 207            Valor mínimo = 0.00383            Valor máximo = 0.11196            Rango = 0.10812            Media aritmética = 0.00806            Error medio estándar = 0.00067            Desviación estándar = 0.00972            Varianza = <math>9.44847 \times 10^{-5}</math>            Desv. estándar poblacional = 0.00972            Varianza poblacional = <math>9.40283 \times 10^{-5}</math>            Coeficiente Skewness = 9.63306            Coeficiente Kurtosis = 95.83590</p>
Gráfico de densidad				Angulares
 <p>Un gráfico de densidad en un sistema de coordenadas 3D (X, Y, Z). El eje Z apunta hacia arriba, Y hacia la derecha y X hacia la izquierda. Se muestra una distribución de puntos que forma una columna vertical estrecha y alta, ubicada cerca del eje Z positivo, indicando una fuerte tendencia positiva en el eje Z.</p>				<p><b>Vectores reales</b></p> <p>Colatitud = 3.19049            Longitud = 111.63643            Media modular = 0.00789</p> <p><b>Vectores unitarios</b></p> <p>Colatitud = 82.40516            Longitud = 30.88569            Media modular = 0.31867            Param. de concentración = 1.46063            Error estándar esférico = 19.96627</p>

## Serie de posición temporal

ZARA00ESP 13462M001

Residual Position Time Series  
(Official EPN Solution C2010)

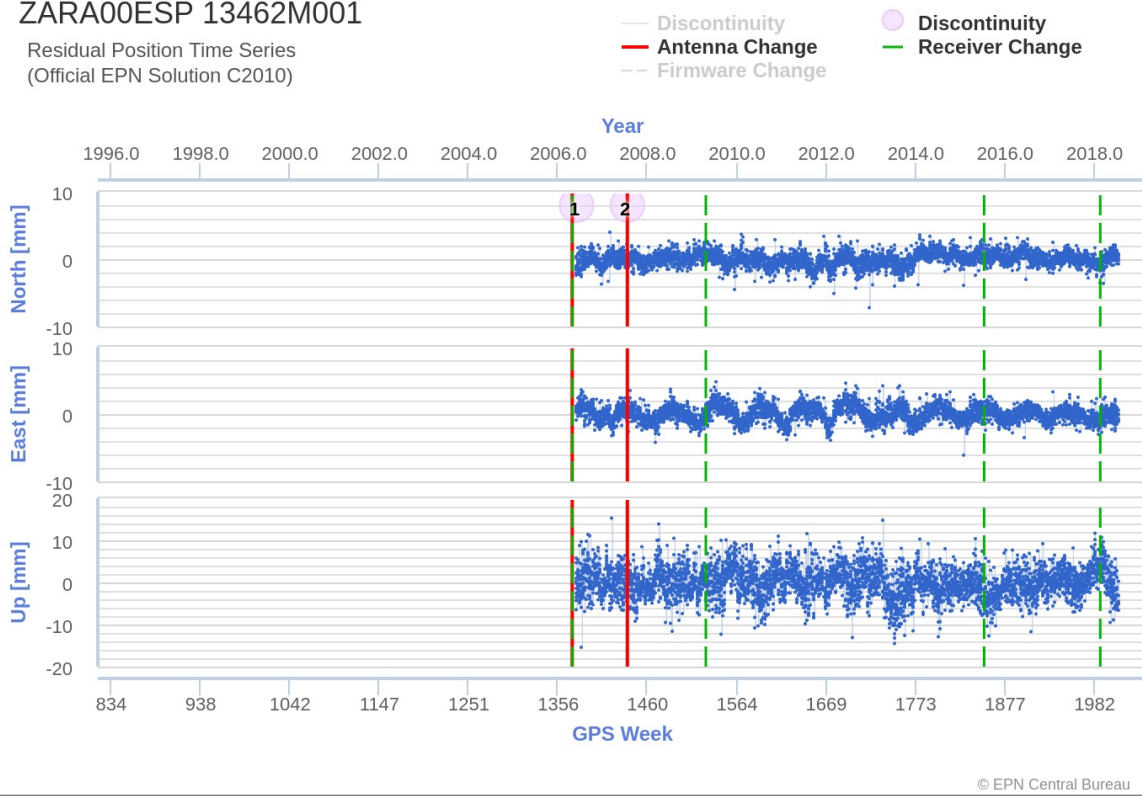
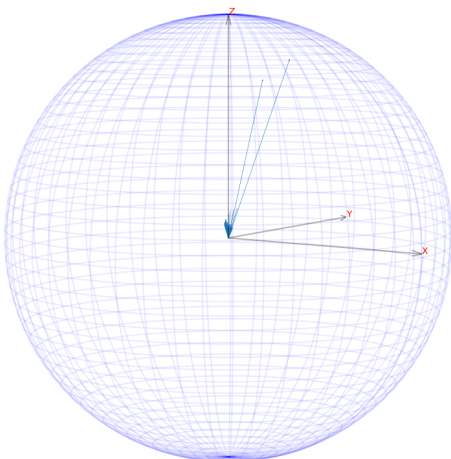
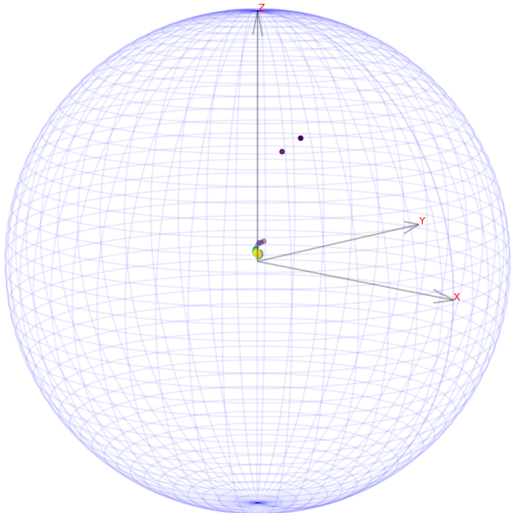


Tabla 8: Análisis de la estación de Zaragoza

La tabla 9, referente a la estación de Santiago, tiene un comportamiento anómalo en comparación con las anteriores, por lo que se ha tenido que rotar el ángulo del gráfico para su correcta visualización, sin embargo, se sigue observando la misma tendencia positiva en Y-Z, negativa en X, y con algunos *outliers para altos valores de Z*.

Posición	Nombre	Primera emisión	Última emisión	Reseña de la estación	
Santiago	<b>CANT</b>	Diciembre 2000	Noviembre 2018	Longitud.....: - 3° 47' 53,03902" Latitud.....: 43° 28' 19,11779" Altitud elipsoidal: 99.245 m.	X.....: 4625924.664 m. Y.....: -307096.774 m. Z.....: 4365771.142 m. Altitud sobre el nivel medio del mar: 49.425 m.
<b>Análisis gráfico</b>				<b>Análisis numérico</b>	
Distribución de módulos y ángulos				Modulares	
				Número de elementos = 201 Valor mínimo = 0.00310 Valor máximo = 0.06003 Rango = 0.05693 Media aritmética = 0.00792 Error medio estándar = 0.00040 Desviación estándar = 0.00580 Varianza = 3.37495 x10 <sup>-5</sup> Desv. estándar poblacional = 0.00580 Varianza poblacional = 3.35816 x10 <sup>-5</sup> Coeficiente Skewness = 6.57006 Coeficiente Kurtosis = 51.00158	
<b>Gráfico de densidad</b>				<b>Angulares</b>	
				<p style="text-align: center;"><b>Vectores reales</b></p> Colatitud = 55.06607 Longitud = 107.30293 Media modular = 0.01265	
				<p style="text-align: center;"><b>Vectores unitarios</b></p> Colatitud = 36.45905 Longitud = 290.41633 Media modular = 0.90280 Param. de concentración = 10.20819 Error estándar esférico = 3.48921	

## Serie de posición temporal



Tabla 9: Análisis de la estación de Santiago

También observamos una discontinuidad temporal en torno al año 2004.

## 6.1 Gráfico georreferenciado

A continuación, en la figura 22, se puede observar, sobre un mapa geográfico, los gráficos de distribución de módulos y ángulos descritos anteriormente. Como se razonaba previamente, aunque las distribuciones sean dispares entre las distintas estaciones, todas ellas muestran una clara tendencia **Y-Z positiva y X negativa**, lo que resulta interesante y se discute más adelante en este documento.

También observamos sobre el mapa la justificación de las estaciones elegidas, cubriendo la periferia de la península con alguna estación en el interior, las de Sonseca y Cáceres.

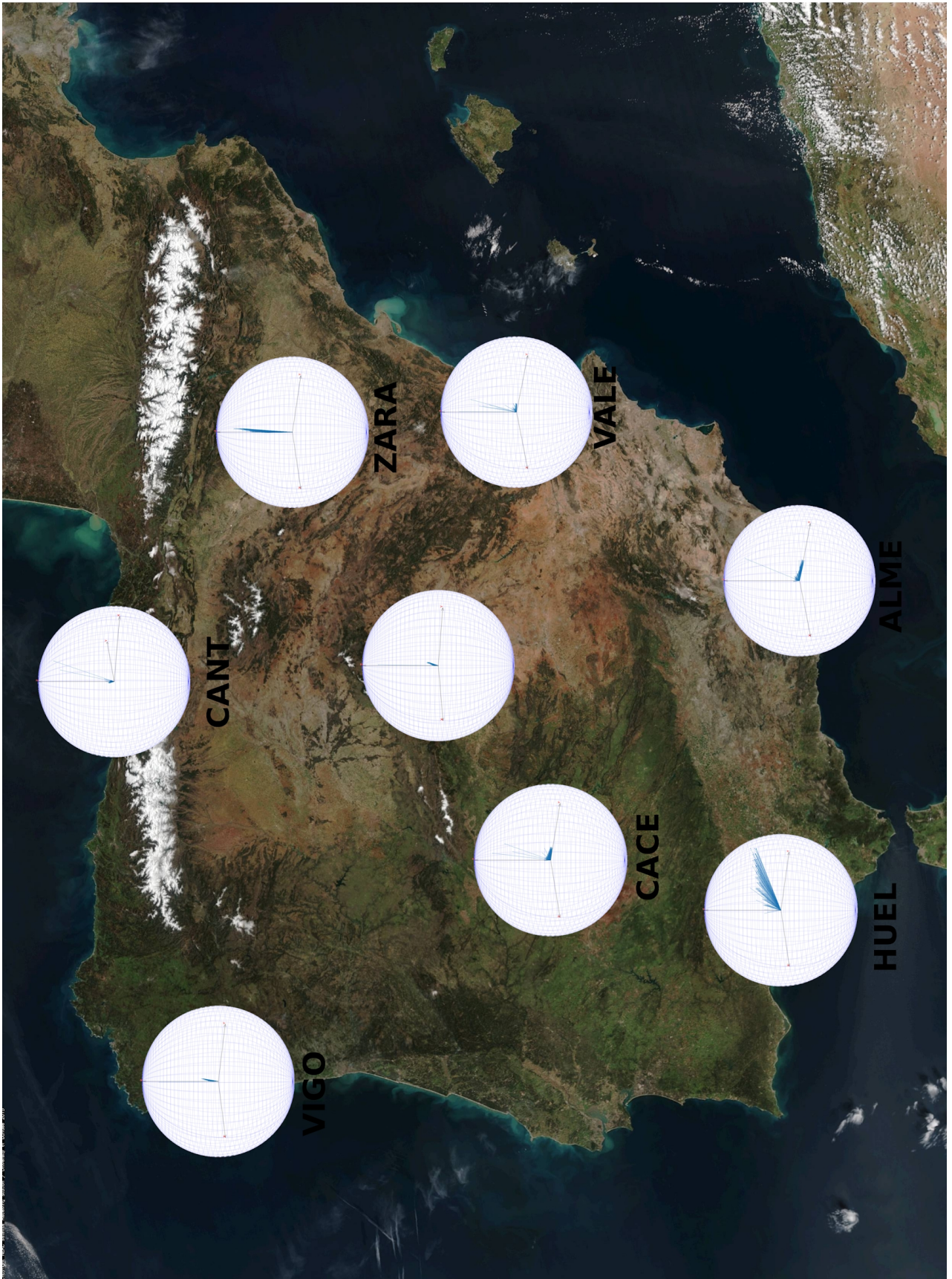


Figura 22: Gráfico georreferenciado





## Capítulo 7: Conclusiones y líneas futuras

### 7.1 Conclusiones generales

Se ha desarrollado una herramienta de análisis estadístico-gráfico para valores tridimensionales que permite su visualización en una escena interactiva con controles básicos: zoom y rotaciones, además permite la exportación en forma de imágenes de la escena generada, se distribuye de forma libre, pudiendo ser utilizada, distribuida y modificada por cualquier persona, se encuentra accesible en este repositorio: <https://github.com/IvanDragoJr/VecStatsGraph3d/> y es de fácil acceso para desarrolladores *Python* mediante la herramienta *PYPI*. Con esto se espera que sea de utilidad para científicos que necesiten de la interpretación de estos datos para llevar a cabo sus investigaciones y publicaciones.

Como prueba de funcionamiento de la librería se han analizado ocho estaciones permanentes GNSS de la red EUREF, obteniendo los estadísticos descritos en la parte de objetivos de este documento, así como mostrando sus gráficas de distribución de módulos y ángulos, y de densidad. Las estaciones elegidas obedecen tanto a cuestiones de datos, se han descartado las de instalación más reciente por no aportar gran cantidad de información, como a cuestiones geográficas, pues se quería tener una red de cobertura de toda la Península Ibérica.

Por otra parte, se ha mejorado el rendimiento de la antigua librería desarrollada en *R* siendo la carga de escenas gráficas significativamente más liviana. Además, se espera que el sistema sea más accesible al público debido a las facilidades del lenguaje *Python*.

Con todo esto, se entiende que han sido superados los objetivos definidos al inicio de este Trabajo de Fin de Grado, que se encuentran descritos en la sección “Objetivos” de este documento.

### 7.2 Líneas futuras

Actualmente la herramienta obtiene los datos de la lectura de un fichero de texto, que se obtiene directamente del *ftp* del IGN y previo proceso se analiza, esto es sin duda, el cuello de botella del rendimiento de la aplicación. Por lo que se entiende que la aplicación podría mejorar significativamente con la inserción de un mecanismo de base de datos, probablemente *NOSQL* que permitiera la rápida lectura de los datos y el almacenamiento de forma más consistente.

Además se observa en los gráficos resultantes la carencia de una funcionalidad para escalar los gráficos, pues aunque se puede hacer zoom sobre la escena, el *grid* esférico mantiene su tamaño dificultando la percepción sobre el resultado.

Por otra parte, como se explica en el anexo ii, la interpretación de las coordenadas ERTF

sigue un proceso semiautomático, si la aplicación va a utilizarse para el análisis de este tipo de datos sería muy recomendable desarrollar un sistema que automatizara por completo este proceso, evitando así errores humanos y consiguiendo un rendimiento necesariamente superior.

## Anexo(i): Manual de usuario

VecStatsGraph3d se encuentra accesible en *PYPI* y es compatible para la versión 2.7 de *Python* por lo que el usuario desarrollador, que quiera acceder a ella deberá integrarla en su proyecto mediante este gestor de librerías, basta con abrir una terminal y ejecutar el siguiente comando:

```
pip install VecStatsGraph3D
```

Una vez finalizado el proceso de instalación la librería estará incluida en el proyecto y se podrá acceder a todas las funcionalidades desarrolladas. Para mostrar esto de forma sencilla se ha desarrollado un *show case* desde el que se pueden probar toda la librería, para acceder a él es necesario importar la librería y ejecutar la función *runShowcase*:

```
from VecStatsGraph.ShowCase import Showcase  
Showcase.runShowcase()
```

Esto ejecutará todas las funciones, por lo que solicitará al usuario la ruta de su fichero de entrada, que deberá contener una lista de vectores representados de la siguiente forma:

- Dos vectores por línea.
- Cada vector representado por un grupo de tres coordenadas.
- Cada coordenada debe estar separada de la siguiente por uno o varios espacios.
- Los puntos deben encontrarse en formato tridimensional.

Se recomienda lanzar este *script* como prueba de que la instalación ha funcionado correctamente y de que se está utilizando una versión de *Python* y *Qt* compatible con la librería.

Una vez instalada la librería se puede acceder a las clases *mánager* que son las que se relacionan con el resto de arquitectura de la librería para llevar a cabo sus funciones, estas son:

- FileManager
- AngleStatisticsManager
- ModuleStatisticsManager
- DensityGraph
- ModuleAngleGraph
- VectorGraph



## Anexo(ii): Interpretación de coordenadas ERTF

El proceso de interpretación de coordenadas ERTF tiene varios pasos:

1. Obtención de las coordenadas del dispositivo.
2. Obtención de los residuos temporales emitidos por el dispositivo.
3. Aplicación de la velocidad y la época a la coordenada original con respecto del residuo concreto y la diferencia temporal entre el momento.

A continuación se explica cómo se ha conseguido realizar estos pasos para obtener todas las coordenadas analizadas:

Las estaciones permanentes de la red EUREF se encuentran descritas en el fichero: [ftp://epncb.oma.be/epncb/station/coord/EPN/EPN\\_A\\_ETRF2014.SSC](ftp://epncb.oma.be/epncb/station/coord/EPN/EPN_A_ETRF2014.SSC) (figura 23) donde podemos encontrar la siguiente información relativa a la época 2010.0 (que se explica más adelante):

```

CLASS A EPN STATION POSITIONS AND VELOCITIES
REFERENCE FRAME: IGS14 AT EPOCH OF 2010.0
CUMULATIVE SOLUTION OF GPSWEEKS [ 0834 - 2010 ]
RELEASE NAME: EPN_A_IGS14_C2010
RELEASED ON 02/10/2018 BY EPN REFERENCE FRAME COORDINATOR (Juliette LEGRAND, ROB, BELGIUM)
DOMES NB. SITE NAME TECH. ID. X/Vx Y/Vy Z/Vz. Sigmas SOLN DATA_START DATA_END REF. EPOCH
CLASS -----m/m/Y-----
-----
10001M007 SMNE GPS SMNE 4201792.244 177945.189 4779286.725 0.001 0.001 0.001 1 00:322:00000 08:118:86370 10:001:00000
10001M007 -.0002 -.0002 -.0001 0.0001 0.0001 0.0001
10001M007 SMNE GPS SMNE 4201792.254 177945.193 4779286.734 0.001 0.001 0.001 2 08:120:00000 09:069:86370 10:001:00000
10001M007 -.0002 -.0002 -.0001 0.0001 0.0001 0.0001
10001M007 SMNE GPS SMNE 4201792.260 177945.188 4779286.735 0.001 0.001 0.001 3 09:071:00000 11:120:86370 10:001:00000
10001M007 -.0002 -.0002 -.0001 0.0001 0.0001 0.0001

```

Figura 23: Formato fichero ERTF

De donde sólo nos interesan las columnas 5, 6 y 7 donde se encuentran representadas, en milímetros, las coordenadas de la estación, además de la columna 2 en la que podemos ver el identificador de la estación, tras obtener esta información tenemos que acudir a la página web:

[http://www.epncb.oma.be/\\_productsservices/timeseries/index.php](http://www.epncb.oma.be/_productsservices/timeseries/index.php)

Y buscar la estación que hemos seleccionado, aquí se muestra una página donde podemos obtener la velocidad de la estación, por ejemplo, para SMNE como se observa en la figura 24.

### Official Station Velocities published by EUREF:

Frame	V <sub>North</sub> [mm/yr]	V <sub>East</sub> [mm/yr]	V <sub>Up</sub> [mm/yr]
IGS14	16.4 ± 0.01	18.0 ± 0.01	-0.1 ± 0.03
ETRF2014	0.1 ± 0.01	-0.2 ± 0.01	-0.1 ± 0.03

Figura 24: Formato velocidades anuales ERTF

Así como su fichero de residuos asociado:

[ftp://epncb.oma.be/pub/station/coord/EPN/Time\\_Series/SMNE\\_10001M007.dat](ftp://epncb.oma.be/pub/station/coord/EPN/Time_Series/SMNE_10001M007.dat)

que tiene el aspecto mostrado en la figura 25.

1088.714	0.00	1.20	0.40	1.00	-5.30	3.80
1088.857	0.00	1.60	0.70	1.20	-1.20	4.80
1089.143	1.70	1.60	0.50	1.20	-4.00	5.00
1089.286	0.60	1.50	1.00	1.10	-1.90	4.50
1089.429	-0.20	1.60	-0.60	1.20	-3.20	4.70
1089.571	1.50	1.80	-0.50	1.30	-1.70	5.30
1089.714	0.50	1.50	1.20	1.10	-3.60	4.40
1089.857	0.70	1.60	0.20	1.20	-2.90	4.70
1090.000	0.10	1.40	0.20	1.10	-1.70	4.20
1090.143	-0.10	1.50	0.40	1.10	-2.40	4.40
1090.286	-0.30	1.80	2.00	1.40	-0.40	5.60
1090.429	-1.20	1.30	0.50	1.00	0.10	3.90
1090.571	0.40	1.40	0.20	1.10	-4.80	4.30
1090.714	-0.80	1.50	0.40	1.10	-2.40	4.40
1090.857	0.50	1.30	-0.20	1.00	-5.90	3.90
1091.000	0.30	1.30	0.00	1.00	-4.60	3.90
1091.143	2.20	1.30	0.30	1.00	-3.10	4.00
1091.286	-0.50	1.40	-0.40	1.10	-2.20	4.40
1091.429	0.30	1.40	0.20	1.10	-5.70	4.30
1091.571	0.20	1.80	0.10	1.40	-2.00	5.50
1091.714	-0.40	1.60	-0.40	1.20	3.10	4.90
1091.857	0.50	1.20	-0.10	1.00	0.30	3.90
1092.000	0.20	1.70	1.00	1.30	2.50	5.00

Figura 25: Formato valores residuales de estaciones ERTF

De donde nos interesan las siguientes columnas:

1. Columna 1: Época en la que se realizó la medición
2. Columna 3: Desvío en x
3. Columna 5: Desvío en y
4. Columna 6: Desvío en z

Una vez conocidos todos estos datos, para cada línea del fichero de residuos aplicamos la fórmula:

$$C(t1)=Ct(0)+V(t1-t0)+R$$

Ecuación 14: Fórmula para el cálculo de la coordenada a partir de los residuos y la velocidad

Siendo C la coordenada, t0 la época de instalación de la estación, t1 la época de la coordenada que se desea obtener, V la velocidad de la estación para la dimensión de la coordenada que se desea obtener y R el desvío de la coordenada.

Las épocas que hemos visto son semanales, en formato gps, lo que es:

- Parte entera: Número de semana desde 1980-01-06.
- Parte decimal: Número del día de la semana empezando por 0 dividido entre 7.

Pero como las velocidades obtenidas son anuales, debemos transformar estas épocas al formato año en GPS:

- Parte entera: Número del año
- Parte decimal: Número del día del año entre 366.

En Internet se encuentran multitud de calculadoras que permiten cambios de formato de este estilo, además, en *Python* se ha encontrado una función que lo realiza automáticamente.





# Bibliografía

## Páginas web y otros recursos

- Aplicaciones de la Estadística Circular a problemas de Ciencias Naturales – Mabel Mena
- <http://www.ign.es>
- <http://www.epncb.oma.be>
- <https://www.datacamp.com/community/tutorials/r-or-python-for-data-analysis>
- <https://www.guru99.com/r-vs-python.html>
- <https://cran.r-project.org/web/packages/VecStatGraphs3D/VecStatGraphs3D.pdf>

## Referencias bibliográficas

[1] Ruiz Cuetos, J C. (2010) Desarrollo de una aplicación práctica sobre estadística esférica. Trabajo de fin de estudios. Ingeniería Informática. Universidad de Extremadura.

[2] Bachelet E. (1981) Circular statistics in biology. London. London Academic Press.

[3] Fisher N.I. (1995) Statistical analysis of circular data. Cambridge. Cambridge University Press.

[4] Fisher, N.I., Lewis, T., Embleton, B.J.J. Statistical Analysis of Spherical Data, Cambridge University Press, 1993. ISBN 0-521-45699-1

[5] Instituto Geográfico Nacional

<http://www.ign.es/web/resources/docs/IGNCnig/GDS-Teoria-Geodesia.pdf>

(Consulta: 18 noviembre 2018)

[6] Royal Observatory of Belgium: EUREF Permanent GNSS Network

[http://www.epncb.oma.be/\\_networkdata/stationmaps.php](http://www.epncb.oma.be/_networkdata/stationmaps.php)

[7] A. Cuartero, M. E. Polo, P. G. Rodríguez, Á. M. Felicísimo y J. C. Ruiz-Cuetos "The Use of Spherical Statistics to Analyze Digital Elevation Models: An example From LIDAR and ASTER GDEM". IEEE 2013

**[8] O'Reilly Company**

<https://www.oreilly.com/library/view/machine-learning-for/9781786469878/f2cbeb0b-4094-4cb9-be54-2d046ff79d7f.xhtml>

(Consulta 18 noviembre 2018)

**[9] The Apache Software Foundation**

<http://www.apache.org/licenses/LICENSE-2.0>

(Consulta 18 noviembre 2018)