

Titulación	Tipo	Curso
2500097 Física	OB	2

Contacto

Nombre: José Flix Molina

Correo electrónico: jose.flix@uab.cat

Idiomas de los grupos

Puede consultar esta información al [final](#) del documento.

Prerrequisitos

No hay requisitos especiales en cuanto a las asignaturas de física. Es conveniente tener aprobado el cálculo 1 del primer semestre. Es recomendable formarse previamente en técnicas de programación en Python (se darán indicaciones y material durante el curso). Otras técnicas matemáticas necesarias son parte de la propia asignatura.

Objetivos y contextualización

En esta asignatura se introducen conceptos básicos de la teoría de probabilidad y estadística, así como unas nociones fundamentales de programación en Python.

La física es en último término una ciencia experimental, basada en las medidas que se realizan en experimentos científicos. Estas medidas están inevitablemente sujetas a errores que se deben cuantificar, a veces en forma de parámetros con sus errores, otras como niveles de confianza, o sobre la base de otros conceptos. En todas las ciencias experimentales se lleva a cabo un proceso similar y la herramienta para hacerlo de manera rigurosa es la rama de las matemáticas denominada Estadística.

Como físicos tenemos que invertir un mínimo de esfuerzo intelectual en entender claramente los conceptos básicos de la Estadística. Esto es válido tanto para los físicos experimentales que realizan las medidas, como para los teóricos que tratan de elaborar modelos o teorías que expliquen las medidas. El lenguaje y los conceptos sobre teoría de probabilidad son también esenciales en algunos de los fundamentos de la física, en concreto en la Mecánica Cuántica y en la Mecánica Estadística. La familiaridad con estos conceptos es necesaria para abordar estas disciplinas.

Para poder afrontar y entender muchos de los conceptos que aprenderemos en clase, haremos unas sesiones prácticas utilizando herramientas actuales de programación en lenguajes modernos, en particular programación en Python sobre Jupyter Notebooks. Siendo conscientes de la poca o nula experiencia en programación, se proporciona material suficiente para poder alcanzar unas nociones elementales de programación con antelación. Las prácticas son ordenadas en orden creciente de dificultad, desde métodos numéricos muy elementales (integración, derivadas), hasta cubrir muchos de los aspectos que se habrán explicado en la parte teórica de la asignatura. Repasaremos las estructuras de datos básicas, así como las estructuras de control fundamentales, y emplearemos las librerías de Python de uso común en problemas físico-matemáticos actuales.

Competencias

- Comunicar eficazmente información compleja de forma clara y concisa, ya sea oralmente, por escrito o mediante TIC, y en presencia de público, tanto a audiencias especializadas como generales.
- Desarrollar estrategias de análisis, síntesis y comunicación que permitan transmitir los conceptos de la Física en entornos educativos y divulgativos.
- Introducir cambios en los métodos y los procesos del ámbito de conocimiento para dar respuestas innovadoras a las necesidades y demandas de la sociedad.
- Razonar críticamente, poseer capacidad analítica, usar correctamente el lenguaje técnico, y elaborar argumentos lógicos.
- Usar instrumentos informáticos (lenguajes de programación y software) adecuados en el estudio de problemas físicos.

Resultados de aprendizaje

1. Analizar y describir con claridad la estrategia al abordar un problema determinado desde el punto de vista numérico.
2. Analizar y describir los problemas físicos desde una perspectiva aproximada modelizando sistemas físicos complejos y solucionándolos aproximadamente.
3. Aplicar los métodos Montecarlo en problemas concretos y resolver algunos de los problemas más habituales.
4. Comunicar eficazmente información compleja de forma clara y concisa, ya sea oralmente, por escrito o mediante TIC, y en presencia de público, tanto a audiencias especializadas como generales.
5. Controlar, profundizando en su análisis, los errores cometidos en los distintos métodos numéricos.
6. Desarrollar estrategias de programación que permitan el uso colaborativo de los programas desarrollados.
7. Elaborar programas en algún lenguaje de programación particular.
8. Identificar situaciones que necesitan un cambio o mejora.
9. Presentar resultados numéricos con precisión, incluyendo el tratamiento estadístico de errores.
10. Razonar críticamente, poseer capacidad analítica, usar correctamente el lenguaje técnico, y elaborar argumentos lógicos.

Contenido

Bloque 1. Teoría de la probabilidad.

Capítulo 1. Probabilidad y Estadística, conceptos básicos.

El concepto de probabilidad y su interpretación. Las escuelas frecuentista y bayesiana. Teoría axiomática de probabilidad. Distribuciones de probabilidad, conceptos de densidad de probabilidad, media, varianza y otros momentos.

Capítulo 2. Probabilidad Condicional. Distribuciones de varias variables

Distribuciones de varias variables, probabilidad condicional, distribuciones marginales. Correlación y covarianza. Cambios de variable.

Capítulo 3. Distribuciones de probabilidad más frecuentes

Distribuciones Binomial, Multinomial, hipergeométrica, de Poisson, Exponencial, Normal y de Gauss. Propiedades de la distribución gaussiana en una y varias variables. La distribución Chi-2, la distribución t-Student, las distribuciones de Cauchy, Gama y de Landau.

Capítulo 4. El método de Monte Carlo

Números aleatorios. Integración por Monte Carlo, los procedimientos de aceptación-rechazo y de transformación de variables.

Bloque 2. Inferencia estadística.

Capítulo 5. Muestreo de una población estadística

Los conceptos de muestra y de estimador. Estimadores de parámetros de una población. Ejemplos elementales.

Capítulo 6. El Método de Máxima Verosimilitud

Concepto de verosimilitud. Varianza de un estimador. La cota de Cramer-Rao-Fréchet. Estimadores de varianza mínima. Aplicaciones del método.

Capítulo 7. Aplicaciones de el Método de Máxima Verosimilitud

Ejemplos de aplicación del método de Máxima Verosimilitud. El método de Mínimos Cuadrados.

Capítulo 8. Contraste de hipótesis

Contraste de bondad de ajuste. Lemma de Neyman-Pearson. Altrest tests. Ejemplos. Errores Estadísticos, Intervalos de Confianza, Límites.

Bloque 3. Programación.

Programación en Python. Uso de Jupyter Notebooks y máquinas virtuales en Google (Google Colab). Estructuras de control básicas en programación.

Uso de las librerías de Python utilizadas con más frecuencia en Data Science.

Sesiones prácticas de métodos numéricos en orden creciente de dificultad: integración numérica (método MonteCarlo), método de transformación, teorema del límite central, intervalos de confianza, factor de correlación, método de máxima verosimilitud, método de Kolmogorov-Smirnov, ...

Actividades formativas y Metodología

Título	Horas	ECTS	Resultados de aprendizaje
Tipo: Dirigidas			
Clases prácticas	13	0,52	1, 2, 4, 9
Clases teóricas	12	0,48	4, 7, 6, 9
Tipo: Autónomas			
Elaboración de trabajos	21	0,84	1, 4, 7, 6
Estudio personal	18,5	0,74	1, 2, 4, 7, 6, 9

Clases teóricas con discusiones en clase.

Realización de problemas ejemplo.

Asignación de problemas y correcciones en clase.

Posibilidad de discusión por medios electrónicos.

Consultas con los profesores de la asignatura.

Nota: se reservarán 15 minutos de una clase dentro del calendario establecido por el centro o por la titulación para que el alumnado rellene las encuestas de evaluación de la actuación del profesorado y de evaluación de la asignatura o módulo.

Evaluación

Actividades de evaluación continuada

Título	Peso	Horas	ECTS	Resultados de aprendizaje
Evaluación y discusión de trabajos	30	3	0,12	4, 7, 6, 8, 9, 10
Examen Parcial Probabilidad y Estadística	35	2,5	0,1	1, 2, 3, 4, 5, 6, 10
Examen Parcial Programación	35	2,5	0,1	1, 2, 4, 6, 10
Examen de repesca	100	2,5	0,1	1, 2, 3, 4, 5, 6, 10

30% Nota: Evaluación y discusión de los problemas asignados. (15% primera parte y 15% segunda parte)

70% Nota: Exámenes Parciales.

La asistencia a las clases prácticas es obligatoria.

Para promediar las dos partes de la asignatura será necesario tener una nota superior a 3,5 puntos en cada parcial. Existe examen de recuperación para aquellos que no superen la asignatura por parciales o quieran subir nota. En el examen de recuperación es posible presentarse únicamente si previamente se han realizado los dos exámenes parciales, y no se tendrán en cuenta las entregas. Es decir, si en la 1ª parte uno tiene un 10 en las entregas y un 4 en el examen, tiene una nota de la primera parte de $0.3 \cdot 10 + 0.7 \cdot 4 = 5.8$, sobre 10. En la recuperación para subir nota se viene a subir el 5.8, no el 4 que se sacó al parcial. Al final, se aplica nota mínima de corte por parte (3.5) para aprobar la asignatura.

Evaluación única

La asistencia a las clases prácticas (de programación) es obligatoria. Al final de cada clase práctica se debe realizar una entrega de los resultados para ser evaluados.

El alumnado que se haya acogido en la modalidad de evaluación única deberá realizar una prueba final que consistirá en un examen que cubrirá tanto la primera como la segunda parte de la asignatura.

La calificación del estudiante será la media ponderada de las tres actividades anteriores, donde el examen de la primera parte supondrá el 35% de la nota, el examen de la segunda parte (programación) el 35% y un 30% los informes /evaluaciones de las prácticas. Se aplica una nota mínima de corte por cada una de las partes (3.5) para aprobar la asignatura, de forma similar a los que realizan la evaluación continua.

Si la nota final no alcanza 5, el estudiante tiene otra oportunidad de superar la asignatura mediante el examen de recuperación que se celebrará en la fecha que fije la coordinación de la titulación. Aquí, la parte de prácticas no es recuperable, y la nota final será la media ponderada que obtengas de la primera y segunda parte, aplicando también el 3.5 como corte por parte, para hacer la media.

Bibliografía

S. Brandt, "Statistical and Computational Methods in Data Analysis", North-Holland Publishing Co., 3rd printing, 1978

Glen Cowan, "Statistical Data Analysis", Oxford Univ. Press, 1998, ISBN 0198501552

Notas detalladas del profesor que se pondrán disponibles en el Campus Virtual.

Software

Para la parte práctica, emplearemos un ordenador, con un navegador para acceder a Google Colab y hacer las prácticas en un entorno Python en la nube.

Lista de idiomas

Nombre	Grupo	Idioma	Semestre	Turno
(PLAB) Prácticas de laboratorio	1	Catalán	segundo cuatrimestre	mañana-mixto
(PLAB) Prácticas de laboratorio	2	Catalán	segundo cuatrimestre	mañana-mixto
(PLAB) Prácticas de laboratorio	3	Catalán	segundo cuatrimestre	mañana-mixto
(PLAB) Prácticas de laboratorio	4	Catalán	segundo cuatrimestre	mañana-mixto
(TE) Teoría	1	Catalán	segundo cuatrimestre	mañana-mixto