

Contenido del curso

Analítica para los negocios · 06327-ECO

Periodo 2026-02

Índice

1. Fundamentos del curso	2
1.1. Semana 01 - Presentación del curso y metodología	2
1.2. Semana 02 - Introducción a LLMs	4
1.3. Semana 03 - Fundamentos de R	5
1.4. Semana 04 - Manipulación y visualización: <code>dplyr</code> + <code>ggplot2</code>	6
2. Proceso analítico y exploración de datos	7
2.1. Semana 05 - Proceso analítico y tipos de analítica	7
2.2. Semana 06 - EDA: fuentes, limpieza y exploración	8
Semana 07 - Examen Parcial	9
3. IA aplicada al análisis de datos	10
3.1. Semana 08 - Claude Code / Cursor / VS Code	10
Semana 09 - Presentación Avance del Proyecto (EDA)	11
4. Fundamentos de Machine Learning	12
4.1. Semana 10 - Fundamentos de Machine Learning	12
5. Aprendizaje supervisado	13
5.1. Semana 11 - Clasificación: pipeline, métricas y modelos base	13
5.2. Semana 12 - Regresión: pipeline, métricas y modelos base	14
6. Aprendizaje no supervisado	15
6.1. Semana 13 - Clustering: fundamentos y métricas	15
Semana 14 - Examen Parcial	16
Semana 15 - Simulacro de la presentación final	17
Semana 16 - Presentación del Proyecto Final	18

Dinámica semanal. En las semanas de contenido, la fundamentación teórica se estudia **antes** de clase mediante un podcast o un video (los enlaces estarán disponibles en Intu) y se evalúa con un **quiz** semanal. La sesión de clase se dedica a la **aplicación** de esa teoría (aplicación guiada / taller).

1. Fundamentos del curso

1.1. Semana 01 - Presentación del curso y metodología

- **1. Presentación del profesor**
 - Perfil y experiencia relevante para el curso.
- **2. Presentación de estudiantes**
 - Ronda rápida: nombre, programa, ¿hace cuánto tiempo vio BA-2?
 - ¿Ha tenido experiencia con datos/R?
- **3. Motivación**
 - ¿Por qué Business Analytics?
 - ¿Qué problemas ayuda a resolver en economía/negocios?
- **4. ¿Qué es Business Analytics? (marco del curso)**
 - Definición operativa.
 - Proceso analítico (alto nivel).
 - Tipos de analítica (alto nivel).
- **5. Contenido del curso (vista general semana a semana)**
 - Recorrido por unidades y semanas.
 - Conexión con el proyecto final.
- **6. Dinámica de aprendizaje (cómo vamos a aprender)**
 - Teoría antes de clase: cada semana de contenido tiene un podcast o video con la fundamentación teórica.
 - Quiz semanal sobre esa teoría (en Intu).
 - La sesión de clase se dedica a la aplicación (aplicación guiada / talleres).
 - Trabajo en clase y trabajo autónomo; expectativas de participación.
- **7. Evaluación y reglas de juego**
 - Componentes: quizzes semanales, talleres en clase, dos exámenes parciales y proyecto final (tres entregas).
 - Todas las semanas de contenido hay quiz sobre el material teórico.
 - No hay recuperación de quizzes ni de talleres; únicamente los exámenes parciales tienen supletorio (según reglamento).
 - Se elimina la peor nota de quizzes y la peor nota de talleres **una sola vez** en el semestre.
 - Entregas del proyecto: quien no presenta una entrega obtiene cero.
- **8. Trabajo final**
 - Datos: una base de datos empresarial con múltiples variables; cada grupo define su propia pregunta de negocio (clasificación, regresión o segmentación).
 - Entregables: (1) pregunta de negocio; (2) presentación del avance/EDA (Semana 09); (3) presentación final (Semana 16).

- La entrega final es una sustentación oral (ya no hay documento escrito) y equivale a un examen parcial: la ausencia solo se acepta mediante supletorio autorizado por la universidad.
- Contenidos mínimos, rúbricas y reglas: documento de lineamientos del proyecto.

■ **9. Política de IA**

- Qué se permite / qué no se permite.
- Cómo citar/explicar uso de IA.
- Enfoque en aprendizaje y trazabilidad.

1.2. Semana 02 - Introducción a LLMs

■ Alcance

- Qué es un LLM (*Large Language Model*) y por qué es relevante para economía y negocios.
- La IA como herramienta transversal del curso: apoyo para R, visualización, proceso analítico y proyecto final.
- Pasar de instrucciones vagas (“hágame esto”) a prompts efectivos que le sacan provecho a la IA.

■ Fundamentación Teórica (podcast + quiz)

- Podcast: *¿Qué es un LLM?*
 - Intuición de cómo funciona un LLM y su arquitectura (alto nivel).
 - Alcances y límites: qué hace bien, dónde se equivoca (alucinaciones) y por qué hay que verificar.
 - Buenas prácticas de uso en contextos académicos y de negocio.
- Material para la sesión:
- Quiz de teoría del podcast (en Intu).

■ Aplicación en clase

- Cómo escribir un buen prompt: contexto, tarea, formato de salida y restricciones; iterar sobre la respuesta.
- Introducción a *skills* (instrucciones reutilizables) para tareas recurrentes.
- Ejercicios de prompts aplicados a tareas del curso (pedir ayuda en R, en visualización, en el proceso analítico).

1.3. Semana 03 - Fundamentos de R

■ Alcance

- Familiarizarse con la interfaz de RStudio: consola, editor (scripts) y paneles principales.
- Usar R como calculadora: operaciones aritméticas, comparaciones y lógica básica.
- Reconocer tipos de datos (numérico, texto, lógico) y valores especiales (NA/NULL).
- Entender qué es un *objeto* en R y cómo asignarlo (operador `<-`).
- Trabajar con estructuras básicas: **vectores**, **matrices** (intro) y **data frames**.
- Introducción mínima a funciones, sistema de ayuda (`?`, `help()`) y carga de librerías.
- Cierre: noción de entorno de trabajo (Environment) y buenas prácticas iniciales (Projects / rutas relativas).

■ Fundamentación Teórica (video + quiz)

- Video de clase: recorrido por la interfaz y los conceptos base.
 - Interfaz de trabajo en RStudio y flujo recomendado: *script* → ejecución → revisión en consola/environment.
 - Tipos de datos y coerción básica (numérico, carácter, lógico; NA/NULL).
 - Objetos en R: asignación, nombres, y funciones para inspección (`class()`, `typeof()`, `str()`).
 - Estructuras de datos: vectores (creación e indexación desde 1), matrices (idea general) y data frames (acceso básico a filas/columnas).
 - Paquetes: diferencia entre instalar (`install.packages`) y cargar (`library/require`); atajo opcional con `pacman::p_load()`.
- Material para la sesión:
- Quiz de teoría (en Intu).

■ Aplicación en clase (aplicación guiada)

- Mini-lab guiado (en parejas):
 - Ejecutar operaciones en consola y luego repetirlas desde un **script**.
 - Crear objetos (`x <- ...`), inspeccionar con `class/typeof/str`.
 - Construir un vector y practicar indexación con `[]`.
 - Crear un data frame pequeño (3 columnas) y realizar accesos básicos (`$` y `[ , ]`).
 - Provocar 1 *warning* y 1 *error* y aprender a interpretarlos.
 - Cargar una librería (o `pacman::p_load`) y verificar que quedó activa.
- Cierre (10 min): conectar Environment con los objetos creados y explicar por qué usaremos Projects/rutas relativas para reproducibilidad.

1.4. Semana 04 - Manipulación y visualización: dplyr + ggplot2

■ Alcance

- Manipulación de **data frames limpios** (sin NA, sin outliers, sin inconsistencias): enfoque en **transformación y resumen**.
- **dplyr** como gramática para transformar datos: filtrar filas, seleccionar columnas, ordenar, renombrar, crear y eliminar variables.
- Resúmenes y descriptivas: estadísticas globales y por grupo (`group_by()` + `summarise()`), conteos con `count()` y cardinalidad con `n_distinct()`.
- **ggplot2** como sistema de construcción de gráficos por capas: `ggplot()` + `aes()`, geometrías (`geom_*()`), etiquetas (`labs()`), temas (`theme_*()`) y escalas (`scale_*()`).
- Producto de la semana: una **tabla analítica de KPIs** y 2–3 gráficos que la comunican.
- Buenas prácticas de comunicación: claridad en ejes, unidades, títulos y leyendas; evitar gráficos confusos o sobrecargados.

■ Fundamentación Teórica (video + quiz)

- Video de clase.
 - Verbo a verbo de **dplyr**: `select()`, `rename()`, `filter()`, `arrange()`, `mutate()`, `summarise()`, `group_by()`, `count()`, `distinct()`.
 - Resúmenes típicos: medidas de centro (media/mediana), dispersión (desviación estándar, mín, máx) y tamaño de muestra por grupo (`n()`).
 - Gramática de gráficos: datos → mapeo estético (`aes`) → geometrías → escalas/etiquetas/tema.
 - Construcción incremental de un gráfico: ejes y estéticos, geometría según el objetivo (`geom_point()`, `geom_histogram()`, `geom_col()`, `geom_line()`), etiquetas y estilo.
 - Opcional (si hay tiempo): facetas para comparar grupos (`facet_wrap()` / `facet_grid()`).
- Material para la sesión:
- Quiz de teoría (en Intu).

■ Aplicación en clase (aplicación guiada)

- Mini-lab guiado usando una base **limpia**:
 - Transformaciones básicas: seleccionar, filtrar, renombrar, ordenar y crear 1–2 variables derivadas.
 - KPIs globales (`summarise()` sin agrupar) y KPIs por grupo (`group_by()` + `summarise()`, incluye `n()`).
 - Construir 2–3 gráficos con el enfoque incremental (gráfico base → geometría → etiquetas → tema).
- Producto mínimo: un script reproducible con la tabla de KPIs y 2–3 visualizaciones con títulos, ejes etiquetados y leyenda clara (cuando aplique).
- Cierre (5–10 min): discusión de qué KPIs son útiles para una decisión de negocio y cómo comunicarlos.

2. Proceso analítico y exploración de datos

2.1. Semana 05 - Proceso analítico y tipos de analítica

■ Alcance

- *Business Analytics* como un proceso que transforma datos en conocimiento accionable para mejorar decisiones.
- Componentes del Business Analytics:
 - Estadística aplicada (patrones, relaciones, significancia).
 - Programación y tecnología (automatización, escalabilidad, reproducibilidad).
 - Ciencia de datos (modelos predictivos y prescriptivos).
 - Enfoque de negocio (la analítica está al servicio de una decisión).
- Tipos de preguntas de negocio y su traducción a tareas analíticas: Supervisada - Clasificación y Predicción. No Supervisada - Segmentación. Mixto - Detección de anomalías. Prescriptiva - Optimización.
- El flujo analítico: de la pregunta a la decisión (visión general del ciclo completo).
- Roles en analítica y su contribución al proyecto: Data Engineer (preparación/ETL y calidad de datos), Data Analyst (EDA, visualización, reportes), Data Scientist (modelado, validación, experimentación) y Business Analyst / Data Translator (traducción técnico–negocio y recomendaciones).

■ Fundamentación Teórica (podcast + quiz)

- Podcast de la clase.
 - ¿Qué es Business Analytics? (definición y componentes).
 - Tareas analíticas típicas y ejemplos de preguntas (clasificación, predicción, segmentación, anomalías, optimización).
 - Flujo analítico: formulación de la pregunta de negocio, recolección de datos, limpieza y transformación, análisis exploratorio y modelado, visualización y comunicación, decisión basada en evidencia.
 - Fases del proceso analítico: preparación de datos (ETL/ELT), exploración (EDA), modelado y resultados (interpretación, comunicación, recomendación y monitoreo).
 - Roles: responsabilidades, entregables típicos y cómo interactúan en un mismo proyecto.
- Material para la sesión:
- Quiz de teoría (en Intu).

■ Aplicación en clase (presentación guiada de una pregunta de negocio)

- Recorrido completo del proceso analítico a partir de una pregunta de negocio, usando como ejemplo presentaciones de proyectos de semestres anteriores.
- Evaluación de preguntas candidatas: de un conjunto de preguntas, identificar cuál está bien formulada (claridad, accionabilidad, factibilidad de datos) y por qué.
- Cierre: cada grupo esboza su propia pregunta de negocio con apoyo de los monitores; este ejercicio es insumo directo para la Entrega 1 del proyecto final.

2.2. Semana 06 - EDA: fuentes, limpieza y exploración

■ Alcance

- Introducción a **fuentes de datos** y calidad de datos en contextos reales (datos no ideales).
- Checklist inicial de revisión: tipos de variables, rangos plausibles, valores imposibles; valores faltantes (NA), categorías inconsistentes, duplicados; outliers y observaciones atípicas.
- Limpieza básica (sin entrar aún en modelado): estandarizar nombres y tipos, tratar valores faltantes (estrategias básicas y transparentes), identificar y manejar outliers (diagnóstico y decisiones justificadas).
- EDA mínima: resúmenes descriptivos y 3–5 visualizaciones para entender el dataset; hallazgos y preguntas abiertas que guían el análisis posterior.
- Producto reproducible: **raw** → **clean** → **analysis-ready** y registro de decisiones.

■ Fundamentación Teórica (podcast + quiz)

- Podcast de la clase (énfasis conceptual; la parte en R se trabaja en la aplicación de clase).
 - Calidad de datos: dimensiones prácticas (completitud, consistencia, validez, unicidad).
 - Revisión estructural: dimensiones, nombres, tipos y diccionario (si existe); chequeos rápidos (conteos, rangos, cardinalidad).
 - Valores faltantes: diagnóstico (conteos/proporciones por variable) y estrategias básicas (eliminar cuando es razonable, imputación simple, indicadores de faltante). Principio: toda decisión debe quedar documentada.
 - Outliers: detección simple (reglas de rango, IQR/boxplot, visualización) y tratamiento (verificar si es error o caso real; recorte/capping o exclusión justificada).
 - EDA: resúmenes globales y por grupo; gráficos exploratorios para distribuciones, comparaciones y relaciones. Conclusión: la EDA produce hipótesis y decisiones de limpieza.
- Material para la sesión:
- Quiz de teoría (en Intu).

■ Aplicación en clase (aplicación guiada)

- Cómo se hace el EDA con R: taller guiado corriendo código sobre un dataset **con problemas intencionales**.
 - Diagnóstico: tipos, NA, duplicados, categorías raras, rangos imposibles.
 - Limpieza: estandarizar nombres/tipos; tratar NA; tratar outliers de forma explícita.
 - EDA mínima: 1 tabla resumen + 3 gráficos clave.
- Producto mínimo: script reproducible de limpieza (con comentarios), dataset “analysis-ready” guardado, log breve de decisiones (qué se cambió y por qué), 3 hallazgos + 3 preguntas abiertas.

Semana 07 - Examen Parcial

3. IA aplicada al análisis de datos

3.1. Semana 08 - Claude Code / Cursor / VS Code

- **Alcance**

- Introducir asistentes de IA para el análisis de datos y la programación: **Claude Code**, **Cursor** y **VS Code** (con asistentes integrados).
- Objetivo central: aprender a hacer con IA lo que ya se hizo “a mano” en las semanas anteriores (manipulación, limpieza, EDA y visualización).
- Conexión con la Semana 02: escribir buenos prompts, ahora en el contexto de código y análisis de datos.
- Uso responsable: la IA acelera el trabajo, pero el estudiante debe entender y validar cada paso (conecta con la política de IA del curso).

- **Fundamentación Teórica (podcast + quiz)**

- Podcast: qué son estas herramientas, cómo funcionan (alto nivel), flujos de trabajo típicos, alcances y límites.
- Material para la sesión:
- Quiz de teoría (en Intu).

- **Aplicación en clase (aplicación guiada)**

- Replicar con apoyo de IA un flujo ya conocido: limpieza + EDA de un dataset (diagnóstico, decisiones documentadas, tabla resumen y gráficos).
- Buenas prácticas: iterar los prompts, revisar el código generado y verificar los resultados contra lo esperado.

Semana 09 - Presentación Avance del Proyecto (EDA)

- Entrega 2 del proyecto final: **presentación del análisis exploratorio (EDA)**.
- Contenido: dada la pregunta de negocio enviada (Entrega 1), cómo se va a responder y qué muestra el EDA de los datos.
- Espacio de retroalimentación del profesor y los monitores de cara a la entrega final.
- Peso: 5 % de la nota total. Quien no presenta obtiene cero.
- Contenido mínimo de la presentación (número de slides, secciones) y rúbrica de evaluación: documento de lineamientos del proyecto.

4. Fundamentos de Machine Learning

4.1. Semana 10 - Fundamentos de Machine Learning

■ Alcance

- Entender el objetivo de Machine Learning como **generalización**: aprender patrones en datos para predecir en datos nuevos.
- Construir el **pipeline** estándar de ML: definir pregunta analítica y variable objetivo (*target*), seleccionar variables predictoras (*features*), separar datos en entrenamiento y prueba (*train/test*), entrenar modelo(s) y ajustar hiperparámetros, evaluar con métricas y comparar contra un baseline, reportar resultados y limitaciones.
- Introducir **validación cruzada** como herramienta para estimar desempeño y reducir sobreajuste.
- Identificar errores comunes: **data leakage**, sobreajuste (*overfitting*) y mala elección de métricas.

■ Fundamentación Teórica (podcast + videos complementarios + quiz)

- Podcast de la clase.
 - Entrenamiento vs prueba: por qué no se evalúa con los mismos datos con los que se entrena.
 - Sobreajuste (*overfitting*) vs subajuste (*underfitting*); baseline como el mínimo razonable para saber si el modelo aporta.
 - Separación de datos: holdout (*train/test*), principios de muestreo y consideración especial con datos temporales (no mezclar pasado y futuro).
 - Validación cruzada: idea de *k*-fold CV y cómo se usa para comparar modelos e hiperparámetros.
 - Riesgos y buenas prácticas: data leakage; el preprocesamiento que se aprende del dataset (escalamiento, imputación) se ajusta **solo** con train y se aplica a test.
- Videos complementarios: construcción de la **matriz de confusión** (métricas de clasificación) y de **MAE/RMSE** (métricas de regresión).
- Material para la sesión:
- Quiz de teoría (en Intu).

■ Aplicación en clase (aplicación guiada)

- Taller guiado con un dataset sencillo (limpio o semi-limpio):
 - Definir *target* y *features*; crear partición train/test (por ejemplo 80/20).
 - Usar predicciones de modelos base previamente entrenados para calcular y comparar métricas de regresión y clasificación.
- Producto mínimo: reporte corto con métrica en test, comparación vs baseline, y 2 riesgos/limitaciones.

5. Aprendizaje supervisado

5.1. Semana 11 - Clasificación: pipeline, métricas y modelos base

■ Alcance

- Definir problemas de **clasificación** (target categórico) y su pipeline completo.
- Métricas de clasificación y decisiones por umbral: matriz de confusión, accuracy (limitaciones), precision/recall y trade-off, (opcional) ROC-AUC como medida agregada.
- Modelos a implementar: **CART** (árbol de clasificación) y **Random Forest** (ensamble), trabajados en la misma clase.
- Buenas prácticas: train/test + validación cruzada para comparar modelos e hiperparámetros.

■ Fundamentación Teórica (video + quiz)

- Video de la clase.
 - Pipeline aplicado a clasificación: partición train/test, CV para selección de hiperparámetros, baseline (clase mayoritaria) como referencia.
 - CART: idea de particiones, profundidad y poda; interpretabilidad vs sobreajuste.
 - Random Forest: idea de ensamble, *bagging*, aleatoriedad en variables; por qué suele mejorar generalización.
 - Consideraciones prácticas: desbalance de clases (intuición) y por qué accuracy engaña.
- Material para la sesión:
- Quiz de teoría (en Intu).

■ Aplicación en clase (aplicación guiada)

- Taller:
 - Definir target y features; crear split train/test.
 - Baseline: clase mayoritaria (y métrica correspondiente).
 - Entrenar CART y evaluar en CV; luego evaluar en test.
 - Entrenar Random Forest (hiperparámetros básicos) y comparar vs CART.
 - Reportar métricas y justificar cuál modelo elegir (según costo de errores).
- Entregable: reporte corto con comparación (baseline vs CART vs RF) + matriz de confusión y recomendación.

5.2. Semana 12 - Regresión: pipeline, métricas y modelos base

■ Alcance

- Problemas de **regresión** (target continuo) y pipeline completo.
- Métricas de regresión: MAE y RMSE (interpretación), comparación contra baseline (media/mediana).
- Modelos a implementar (enfoque práctico y comparativo): **regresión lineal regularizada** (Lasso; Ridge opcional), **árbol de regresión (CART)**, **Random Forest** o **XGBoost** (uno de los dos, según tiempo).
- Objetivo pedagógico: entender el trade-off entre interpretabilidad, desempeño y riesgo de sobreajuste.

■ Fundamentación Teórica (video + quiz)

- Video de la clase.
 - Pipeline aplicado a regresión: train/test + CV para selección (p. ej., lambda en Lasso), baseline (media/mediana) como referencia.
 - Lasso (y Ridge opcional): idea de regularización y por qué ayuda a generalizar; selección de variables (intuición en Lasso).
 - CART de regresión y ensambles: árboles (no linealidades e interacciones); Random Forest/XGBoost y por qué mejoran desempeño (visión general).
 - Interpretación: error en test, diagnóstico simple (residuales), y relevancia de variables (alto nivel).
- Material para la sesión:
- Quiz de teoría (en Intu).

■ Aplicación en clase (aplicación guiada)

- Taller:
 - Definir target continuo y features; split train/test.
 - Baseline: media/mediana y cálculo de MAE/RMSE.
 - Entrenar Lasso (CV para lambda) y evaluar en test.
 - Entrenar CART de regresión y comparar.
 - Entrenar Random Forest o XGBoost y comparar desempeño vs modelos anteriores.
- Entregable: tabla comparativa de MAE/RMSE (baseline vs modelos) + recomendación final (con trade-offs).

6. Aprendizaje no supervisado

6.1. Semana 13 - Clustering: fundamentos y métricas

■ Alcance

- Introducir clustering como herramienta para **segmentación** (sin target).
- Aplicar **k-means** como algoritmo principal.
- Conocer alternativas (alto nivel) y cuándo usarlas: clustering jerárquico (dendrograma), (opcional) DBSCAN para clusters de forma irregular/outliers.
- Métricas y selección del número de clusters: elbow / within-cluster sum of squares, silhouette, estabilidad e interpretabilidad.
- Producto: segmentos + **perfilamiento** (tablas y gráficos) para decisiones de negocio.

■ Fundamentación Teórica (video + quiz)

- Video de la clase (el componente visual es clave para entender cómo se forman los clusters).
 - Intuición: distancia/similitud, y por qué el escalamiento importa.
 - k-means: idea de centroides, sensibilidad a escala e inicialización, qué optimiza (compactación intra-cluster).
 - Elección de k : elbow y silhouette (interpretación).
 - Interpretación: un cluster solo sirve si puede describirse y accionarse.
- Material para la sesión:
- Quiz de teoría (en Intu).

■ Aplicación en clase (aplicación guiada)

- Taller:
 - Preparar variables numéricas (selección + escalamiento).
 - Estimar k-means para varios k .
 - Elegir k con elbow/silhouette (y justificar).
 - Asignar etiquetas de cluster y perfilar: promedios/medianas por cluster, tamaño de cluster, 2 visualizaciones (p. ej., scatter + boxplots por cluster).
- Entregable: tabla de perfilamiento + 2 gráficos + 3 insights accionables por segmento.

Semana 14 - Examen Parcial

Semana 15 - Simulacro de la presentación final

- Cada grupo presenta una versión preliminar de su sustentación del proyecto final.
- Espacio de retroalimentación de fondo y forma (profesor y monitores) antes de la presentación definitiva.
- Objetivo: llegar a la Semana 16 con la presentación ajustada.

Semana 16 - Presentación del Proyecto Final

- Sustentación oral del proyecto final (Entrega 3). Ya no hay entrega de documento escrito: la evaluación final es la presentación.
- Equivale a un examen parcial: la asistencia es **obligatoria** y la ausencia solo se acepta mediante supletorio autorizado por la universidad.
- El formato de la presentación es libre; los criterios de evaluación están en la rúbrica del documento de lineamientos del proyecto.