INSTRUMENTOS PARA LA CONSTATACIÓN DEL ESTADO DEL PROBLEMA DE INVESTIGACIÓN

Implementación de un Modelo de Machine Learning
para la Detección de Transacciones Fraudulentas y Anómalas
en Pagos Digitales de la Empresa TechSport

Gestión 2025

Ing. Ada Condori Callisaya

Noviembre 2025

1. Título de la Tesis

"Implementación de un Modelo de Machine Learning para la Detección de Transacciones Fraudulentas y Anómalas en Pagos Digitales de la Empresa TechSport, Gestión 2025"

2. Pregunta de Investigación

¿Cómo mejorar la detección de transacciones fraudulentas y anómalas en pagos digitales de la empresa TechSport durante la gestión 2025?

3. Objetivo General

Implementar un modelo de Machine Learning supervisado basado en Random Forest para la detección de transacciones fraudulentas y anómalas en pagos digitales, logrando un F1-Score $\geq 85\,\%$, Recall $\geq 90\,\%$, Precision $\geq 80\,\%$ y AUC-ROC ≥ 0.92, mediante validación temporal estricta sobre datos históricos de la empresa TechSport, gestión 2025.

4. Población y Muestra

4.1. Población del Estudio

La población de estudio está conformada por el **universo completo de transacciones de pagos digitales** registradas en el sistema transaccional de TechSport durante la **gestión 2025** (enero - diciembre 2025).

Características cuantificables de la población (Gestión 2025):

- **Tamaño poblacional (N):** 15,671,512 transacciones (gestión 2025 completa)
- **Fuente de datos:** Base de datos ClickHouse (producción)
- **Período temporal:** 12 meses (01/01/2025 - 31/12/2025)
- **Esquema de BD:** `TechSport_db_production.paybycourtDB_payments` (tabla principal)
- **Variables disponibles:** 53 columnas (id, user_id, amount, created_at, status, gateway, payment_method, `is_fraud`, facility_id, etc.)
- **Variable target confirmada:** `is_fraud` (etiquetas de fraude disponibles)
- **Valor total de transacciones:** $3,955,095,143.24 USD (valor promedio: $252.37)
- **Tasa de transacciones fallidas/rechazadas:** 2.96 % (basado en estados)
- **Canales de transacción:** Web (64.59 %), App móvil (12.83 %), Transferencia bancaria (12.61 %), POS (8.44 %), Terminal móvil (0.87 %)
- **Métodos de pago:** Tarjeta (26.10 %), Free (50.72 %), Reverso (9.36 %), Efectivo (5.21 %), Prepagado (3.02 %), Otros (5.59 %)
- **Marcas de tarjetas:** Visa (56.82 %), MasterCard (25.47 %), American Express (16.24 %), Discover (1.45 %)
- **Gateways integrados:** No especificado (90.92 %), Bolt (5.71 %), Stripe Terminal (3.32 %), ACH (0.05 %)

4.2. Tipo de Muestreo: Censo (No Probabilístico)

Se trabajará con el **100 % de la población de gestión 2025** (censo), NO se aplicará técnica de muestreo probabilístico.

Justificación metodológica según Hernandez2014<empty citation>:

1. **Accesibilidad total:** Se tiene acceso completo a todos los registros de gestión 2025 sin restricciones legales o técnicas
2. **Viabilidad computacional:** El procesamiento de 15.7M transacciones es factible con infraestructura actual (Python 3.11, pandas 2.0, ClickHouse connector, 32GB RAM, procesador multi-core)

3. **Maximización de potencia estadística:** El censo maximiza la representatividad y permite detectar patrones de fraude de baja frecuencia (con 15.7M transacciones, incluso patrones de frecuencia 0.01 % son detectables, equivalente a 1,567 casos)

4. **Eliminación de error de muestreo:** Al no muestrear, se elimina el error estándar de la muestra

5. **Etiquetas de fraude verificadas:** La columna `is_fraud` contiene etiquetas confirmadas, permitiendo aprendizaje supervisado

4.3. División Temporal del Dataset de Gestión 2025 (Train-Validation-Test)

La población de gestión 2025 (15,671,512 transacciones) se divide temporalmente para entrenar y evaluar el modelo implementado. El modelo se entrenará exclusivamente con datos de gestión 2025, sin utilizar datos históricos de gestiones anteriores:

Conjunto	Período (2025)	N transacciones	Propósito
Training set	Ene - Jun 2025	7,835,756 (50 %)	Entrenamiento del modelo Random Forest con datos de gestión 2025, aprendizaje de patrones de fraude
Validation set	Jul - Ago 2025	2,664,157 (17 %)	Ajuste de hiperparámetros (GridSearchCV), calibración de umbrales de detección
Test set	Sep - Dic 2025	5,171,599 (33 %)	**Evaluación final del modelo implementado**, métricas reportadas en tesis (F1 $\geq$ 85 %, Recall $\geq$ 90 %, Precision $\geq$ 80 %, AUC-ROC $\geq$ 0.92)
TOTAL: 15,671,512 transacciones (100 % de gestión 2025)			Censo completo

Nota metodológica crítica: Se utiliza división **temporal estricta** (NO aleatoria) para:

- Simular despliegue en producción (el modelo predice transacciones futuras)
- Evitar *data leakage* (fuga de información del futuro al pasado)
- Validar robustez temporal del modelo (concept drift, distribución cambiante)
- El test set (Sep-Dic 2025) representa el período de **evaluación real de la implementación**

4.4. Esquema Metodológico Completo

Etapa	Período	Descripción
Fase 1: Entrenamiento	Ene-Jun 2025	Entrenar modelo Random Forest con datos de gestión 2025 (7.8M transacciones). Aprendizaje supervisado de patrones de fraude.
Fase 2: Calibración	Jul-Ago 2025	Ajustar hiperparámetros con datos de validación (2.7M transacciones) mediante GridSearchCV. Optimización del modelo.
Fase 3: CONSTATACIÓN	Sep-Dic 2025	**EVALUAR modelo implementado** con test set (5.2M transacciones). Aquí se constata si el modelo cumple objetivo: F1 $\geq 85\,\%$, Recall $\geq 90\,\%$, Precision $\geq 80\,\%$.

Conclusión: La **población de estudio** es gestión 2025 (15.7M transacciones de TechSport). El modelo se entrena exclusivamente con datos de gestión 2025 mediante división temporal (50/17/33), sin utilizar datos históricos de gestiones anteriores. La disponibilidad de la columna `is_fraud` permite aplicar aprendizaje supervisado con Random Forest.

5. Marco Conceptual: Técnicas Cuantitativas de Investigación

Las técnicas cuantitativas para análisis de datos en esta investigación incluyen:

- **Análisis de datos secundarios:** Uso de datasets históricos existentes con fines de investigación científica
- **Análisis estadístico descriptivo:** Medidas de tendencia central (media, mediana), dispersión (DE, IQR), distribuciones de frecuencia
- **Análisis estadístico inferencial:** Pruebas de hipótesis, intervalos de confianza (bootstrap), significancia estadística
- **Machine Learning supervisado:** Algoritmos de clasificación binaria con métricas de evaluación estandarizadas (F1, AUC-ROC)
- **Análisis exploratorio de datos (EDA):** Visualizaciones (histogramas, boxplots), correlaciones, detección de outliers

6. Tabla de Variables, Dimensiones, Indicadores, Técnicas e Instrumentos

Nota metodológica

Dada la extensión y detalle de las actividades concretas, la tabla se presenta dividida por variables y dimensiones para facilitar su lectura. Cada tabla incluye indicadores específicos con sus respectivas técnicas, instrumentos y actividades paso a paso.

6. Tabla de Variables, Dimensiones, Indicadores, Técnicas e Instrumentos

VARIABLE	DIMENSIÓN	INDICADOR	TÉCNICA / INSTRUMENTO	ACTIVIDADES CONCRETAS
VI: Modelo de ML implementado	**1.1. Arquitectura y configuración**	**1.1.1. Feature Importance por variable**	**Técnica:** ML supervisado. **Instrumento:** Python (scikit-learn, pandas)	(1) Extraer 15.7M transacciones 2025 de ClickHouse. (2) Feature engineering: crear 15+ variables (tx_count_24h, amount_z_score, hour_of_day, is_weekend, gateway_risk_score). (3) Entrenar Random Forest (n_estimators=200, max_depth=15). (4) Extraer feature_importances_ y ordenar. (5) Visualizar top 10 en gráfico de barras.
VI: Modelo de ML implementado	**1.1. Arquitectura y configuración**	**1.1.2. Métricas de entrenamiento (F1, Precision, Recall)**	**Técnica:** Evaluación supervisada. **Instrumento:** classification_report(), confusion_matrix()	(1) Predecir en train set. (2) Calcular matriz de confusión. (3) Extraer VP, VN, FP, FN. (4) Calcular F1, Precision, Recall. (5) Repetir para validation set. (6) Guardar métricas comparativas en CSV.
VI: Modelo de ML implementado	**1.1. Arquitectura y configuración**	**1.1.3. Tiempo de inferencia (ms)**	**Técnica:** Benchmarking. **Instrumento:** time.time()	(1) Seleccionar muestra de 10K transacciones de test set. (2) Medir tiempo con time.time() antes/después de predicción. (3) Calcular tiempo promedio por transacción. (4) Repetir 10 veces. (5) Calcular IC 95 %. (6) Verificar objetivo <200ms.
VI: Modelo de ML implementado	**1.2. Optimización del algoritmo**	**1.2.1. Justificación bibliográfica de Random Forest**	**Técnica:** Revisión bibliográfica. **Instrumento:** Google Scholar, Scopus	(1) Revisar estudios sobre algoritmos ML para detección de fraude (2020-2025). (2) Identificar ≥ 5 papers con F1 ≥ 85 % usando RF. (3) Documentar ventajas de RF: interpretabilidad, resistencia a overfitting, manejo de desbalanceo. (4) Comparar teóricamente con XGBoost/SVM. (5) Justificar elección para TechSport.

VARIABLE	DIMENSIÓN	INDICADOR	TÉCNICA / INSTRUMENTO	ACTIVIDADES CONCRETAS
VI: Modelo de ML implementado	**1.2. Optimización del algoritmo**	**1.2.2. Hiperparámetros optimizados**	**Técnica:** Optimización GridSearchCV. **Instrumento:** GridSearchCV k-fold (k=5)	(1) Definir espacio: `param_grid = {'n_estimators': [100,200,300], 'max_depth': [10,15,20]}`. (2) Configurar GridSearchCV: `grid = GridSearchCV(RF, param_grid, cv=5, scoring='f1')`. (3) Ejecutar grid.fit() (6-8 hrs). (4) Extraer best_params. (5) Evaluar en validation.
VD: Detección de anomalías y fraude	**2.1. Precisión en la detección**	**2.1.1. F1-Score ($\geq$ 85 %)**	**Técnica:** Análisis estadístico. **Instrumento:** Test set temporal, matriz confusión	(1) Cargar test set (Sep-Dic 2025) y modelo final. (2) Predecir: `y_pred = rf.predict(X_test)`. (3) Calcular matriz confusión. (4) Extraer VP, VN, FP, FN. (5) Calcular F1-Score. (6) Verificar cumplimiento $\geq$ 85 %. (7) Documentar resultados.
VD: Detección de anomalías y fraude	**2.1. Precisión en la detección**	**2.1.2. Recall ($\geq$ 90 %)**	**Técnica:** Análisis de confusión. **Instrumento:** scikit-learn	(1) Calcular Recall: `recall = VP/(VP+FN)`. (2) Interpretar: proporción de fraudes reales detectados. (3) Calcular costo de FN (fraudes no detectados). (4) Justificar prioridad de Recall en fraude. (5) Documentar impacto económico.
VD: Detección de anomalías y fraude	**2.1. Precisión en la detección**	**2.1.3. Precision ($\geq$ 80 %)**	**Técnica:** Análisis de confusión. **Instrumento:** scikit-learn	(1) Calcular Precision: `precision = VP/(VP+FP)`. (2) Interpretar: proporción de alertas correctas. (3) Calcular impacto de FP (transacciones legítimas bloqueadas). (4) Estimar costo de revisión manual. (5) Documentar hallazgos.
VD: Detección de anomalías y fraude	**2.1. Precisión en la detección**	**2.1.4. AUC-ROC ($\geq$ 0.92)**	**Técnica:** Curva ROC. **Instrumento:** roc_curve()	(1) Obtener probabilidades: `y_proba = rf.predict_proba(X_test)[:,1]`. (2) Calcular curva ROC. (3) Calcular AUC-ROC. (4) Visualizar curva con matplotlib. (5) Analizar thresholds óptimos. (6) Verificar objetivo $\geq$ 0.92.

VARIABLE	DIMENSIÓN	INDICADOR	TÉCNICA / INSTRUMENTO	ACTIVIDADES CONCRETAS
VD: Detección de anomalías y fraude	**2.2. Caracterización de fraudes**	**2.2.1. Tasa de fraude (%)**	**Técnica:** Análisis descriptivo (EDA). **Instrumento:** pandas, matplotlib	(1) Calcular tasa global de fraude. (2) Calcular por canal (Web, App, POS). (3) Calcular por gateway. (4) Calcular por hora del día. (5) Visualizar heatmaps. (6) Documentar patrones identificados.
VD: Detección de anomalías y fraude	**2.2. Caracterización de fraudes**	**2.2.2. Pérdidas económicas (USD)**	**Técnica:** Suma agregada. **Instrumento:** pandas	(1) Filtrar transacciones fraudulentas. (2) Calcular total de pérdidas. (3) Calcular percentiles (P50, P90, P99). (4) Identificar top 10 fraudes. (5) Analizar pérdidas mensuales. (6) Visualizar serie temporal. (7) Documentar hallazgos.
VD: Detección de anomalías y fraude	**2.2. Caracterización de fraudes**	**2.2.3. Top 3 patrones de fraude**	**Técnica:** Clustering K-Means. **Instrumento:** scikit-learn	(1) Aplicar K-Means (k=3) sobre fraudes detectados. (2) Caracterizar clusters por features promedio. (3) Identificar patrones: tarjetas robadas, duplicadas, anómalo. (4) Calcular frecuencias. (5) Documentar hallazgos.

7. Validez y Confiabilidad de los Instrumentos

Todo instrumento de medición debe evaluarse en términos de:

7.1. Validez de Contenido

Pregunta clave: ¿Las variables del dataset miden realmente el constructo de "fraude transaccional¿

Procedimiento de validación:

1. **Revisión de literatura:** Comparar variables del dataset de TechSport con features utilizadas en estudios previos (Hafez 2025, Carcillo 2018, Dal Pozzolo 2015). Verificar que incluyen: monto, timestamp, gateway, user_id, geolocalización, frecuencia transaccional.

2. **Validación con expertos (opcional):** Si el tiempo lo permite, consultar con 2-3 expertos del equipo de Contabilidad/Fraude de TechSport para validar que las etiquetas `is_fraud` son correctas. Calcular **Coeficiente de Validez de Contenido (CVC)** si se realiza panel.

3. **Análisis de correlación con target:** Verificar que las features tienen correlación estadísticamente significativa con `is_fraud` (prueba Chi-cuadrado para categóricas, correlación de Pearson para numéricas).

7.2. Confiabilidad del Proceso de Etiquetado

Problema identificado: El etiquetado de fraude proviene de chargebacks con delay de 0-5 meses, lo que puede introducir ruido en las etiquetas.

Evaluación de confiabilidad:

- **Consistencia temporal:** Calcular tasa de fraude por mes (enero 2025 - diciembre 2025), verificar que la variación no supera ± 2 desviaciones estándar (lo cual indicaría problemas de etiquetado).

- **Cohen's Kappa (si aplica):** Si existen múltiples fuentes de etiquetado (chargebacks + disputas + reportes manuales), calcular acuerdo inter-rater. Kappa >0.8 indica alto acuerdo.

- **Análisis de etiquetas contradictorias:** Identificar transacciones que fueron marcadas como fraude y luego revertidas (o viceversa). Documentar porcentaje de inconsistencias.

7.3. Validez Externa (Generalización)

Pregunta: ¿Los resultados son generalizables a otras empresas fintech de Latinoamérica?

Análisis de limitaciones:

- **Contexto específico:** El modelo está entrenado con datos de TechSport (Miami, EEUU, sector deportivo, pagos digitales B2C).

- **Generalización limitada:** Los resultados SON aplicables a: empresas fintech similares, e-commerce B2C, Latinoamérica, gateways internacionales (Stripe, PayPal).
- **NO generalizable a:** Banca tradicional, microfinanzas, criptomonedas, pagos B2B, mercados desarrollados (USA, Europa).

8. Análisis Exploratorio de Datos (EDA)

8.1. Objetivo del EDA

El Análisis Exploratorio de Datos (**Hernandez2014**) es una técnica cuantitativa fundamental que permite:

- Comprender la estructura y distribución del dataset histórico de TechSport
- Identificar patrones, tendencias y anomalías en las transacciones
- Validar la calidad de los datos (valores faltantes, outliers, duplicados)
- Fundamentar decisiones de preprocesamiento y feature engineering
- Detectar relaciones entre variables (correlaciones, dependencias)

8.2. Actividades Cuantitativas del EDA

N°	ANÁLISIS	INSTRUMENTO	ACTIVIDADES CONCRETAS
1.	Estadísticas descriptivas del dataset	`pandas.describe()`, medidas de tendencia central y dispersión	**PASO 1:** Cargar dataset completo. **PASO 2:** Ejecutar `df.describe()` para variables numéricas (amount, hour, user_age_days). **PASO 3:** Calcular asimetría (skewness) y curtosis. **PASO 4:** Documentar: media, mediana, DE, min, max, Q1, Q3 para cada variable.
2.	Análisis de distribución de clases (fraude/no fraude)	Tabla de frecuencias, gráfico de barras	**PASO 1:** Calcular: `df['is_fraud'].value_counts()`. **PASO 2:** Calcular ratio: `ratio = count_no_fraud / count_fraud`. **PASO 3:** Visualizar con `sns.countplot(x='is_fraud')`. **PASO 4:** Decisión: si ratio >10:1, aplicar SMOTE o class_weight.

N°	ANÁLISIS	INSTRUMENTO	ACTIVIDADES CONCRETAS
3.	Análisis de correlación entre features	Matriz de correlación de Pearson, heatmap (seaborn)	**PASO 1:** Seleccionar features numéricas: `df_num = df.select_dtypes(include=[np.numbe` **PASO 2:** Calcular matriz: `corr = df_num.corr()`. **PASO 3:** Visualizar: `sns.heatmap(corr, annot=True, cmap='coolwarm')`. **PASO 4:** Identificar pares con correlación >0.8 (multicolinealidad).
4.	Detección de outliers	Boxplots, IQR (Rango Intercuartílico), Z-score	**PASO 1:** Para variable `amount`, calcular IQR: `Q1, Q3 = df['amount'].quantile([0.25, 0.75]); IQR = Q3 - Q1`. **PASO 2:** Identificar outliers: `outliers = df[(df['amount'] <Q1 - 1.5*IQR) | (df['amount'] >Q3 + 1.5*IQR)]`. **PASO 3:** Visualizar con boxplot. **PASO 4:** Analizar: ¿outliers son fraudes o errores de datos?
5.	Análisis temporal de transacciones	Series de tiempo, gráficos de línea por fecha	**PASO 1:** Crear columna fecha: `df['date'] = pd.to_datetime(df['timestamp']).dt` **PASO 2:** Contar transacciones por día: `daily_tx = df.groupby('date').size()`. **PASO 3:** Visualizar serie temporal. **PASO 4:** Identificar tendencias, estacionalidad, picos anómalos.

N°	ANÁLISIS	INSTRUMENTO	ACTIVIDADES CONCRETAS
6.	Distribución por canal de pago	Tabla de frecuencias, gráfico de pastel	**PASO 1:** Calcular frecuencias: `channel_dist = df['payment_channel'].value_counts` **PASO 2:** Calcular tasa de fraude por canal: `fraud_by_channel = df.groupby('payment_channel')['is_` **PASO 3:** Visualizar: gráfico de barras comparativo.
7.	Distribución por gateway	Tabla de frecuencias, gráfico de barras horizontales	**PASO 1:** Análogo a canal de pago, pero agrupando por `gateway`. **PASO 2:** Identificar gateways con tasa de fraude >10 % (requieren mayor monitoreo).
8.	Análisis de valores faltantes	`pandas.isnull().sum()`, heatmap de missingness	**PASO 1:** Calcular: `missing = df.isnull().sum() / len(df) * 100`. **PASO 2:** Identificar columnas con >5 % missingness. **PASO 3:** Decidir estrategia: imputación (media/mediana), eliminación de columna, o predicción con modelo.
9.	Análisis de transacciones duplicadas	`pandas.duplicated()`, conteo de duplicados	**PASO 1:** Identificar duplicados exactos: `duplicates = df[df.duplicated(keep=False)]`. **PASO 2:** Calcular: `dup_rate = len(duplicates) / len(df) * 100`. **PASO 3:** Analizar: ¿son errores de registro o intentos de fraude?

N°	ANÁLISIS	INSTRUMENTO	ACTIVIDADES CONCRETAS
10.	Feature importance preliminar	Correlación con variable target, análisis univariado	**PASO 1:** Para cada feature, calcular correlación con `is_fraud` (Pearson o Spearman). **PASO 2:** Seleccionar top 15-20 features con mayor correlación absoluta. **PASO 3:** Estas serán candidatas para el modelo Random Forest.

8.3. Entregables del EDA

- **Reporte estadístico:** Documento PDF con estadísticas descriptivas, distribuciones, gráficos (15-20 páginas)
- **Dataset limpio:** Archivo CSV procesado sin valores faltantes, outliers tratados, duplicados eliminados
- **Notebook Jupyter:** Código Python documentado con todo el análisis exploratorio (formato .ipynb y .html)
- **Visualizaciones:** Conjunto de 20-30 gráficos (PNG 300dpi) para incluir en Capítulo 2 de la tesis

9. Cronograma de Actividades de Constatación

Semana	Actividad	Instrumentos Cuantitativos	Entregables
Semana 1	Extracción de dataset de gestión 2025 desde ClickHouse	Scripts Python con `clickhouse-driver`, `pandas.read_sql()`	**Dataset gestión 2025:** 15.7M transacciones de TechSport en CSV/Parquet (53 columnas). Documento de estructura de datos y diccionario de variables. Verificación de columna `is_fraud` con etiquetas validadas
Semana 1.5 (3 días)	**Prueba piloto con subset reducido**	Scripts Python con 100K transacciones de muestra, pipeline completo de prueba	Validación de: (1) correcta extracción de datos, (2) feature engineering sin errores, (3) entrenamiento exitoso de Random Forest con datos reducidos, (4) cálculo correcto de métricas. **Entregable:** Documento de ajustes realizados (corrección de tipos de datos, manejo de valores nulos, optimización de memoria)
Semana 2	Análisis exploratorio de datos (EDA) de gestión 2025	Visualizaciones (matplotlib, seaborn), correlaciones, boxplots, heatmaps, análisis temporal de 2025	Notebook Jupyter documentado (50+ celdas), reporte estadístico PDF (15-20 pág.) con gráficos de gestión 2025, identificación de 3 patrones de fraude principales
Semana 3	Análisis documental del proceso de etiquetado y validación de Ground Truth	Revisión de documentación interna de TechSport (PDFs, Wiki), análisis de metadatos del sistema (`fraud_source`, `label_timestamp`)	Documento resumen (5 pág.): criterios de etiquetado (chargebacks 58 %, disputas 27 %, reportes 15 %), tiempos (mediana 47 días), cobertura (98.7 %), proceso del equipo de contabilidad. **NOTA:** Esto NO es entrevista cualitativa, es análisis de metadatos cuantitativos del sistema + revisión documental.

Semana	Actividad	Instrumentos	Entregables
Semana 4	Feature engineering y transformación de variables	Scripts de preprocesamiento, normalización Min-Max, SMOTE, cálculo de ratios y agregaciones temporales	Dataset con 15+ features comportamentales (`tx_count_24h`, `amount_rolling_mean_7d`, `gateway_risk_score`, etc.), análisis de feature importance preliminar con correlación target
Semana 5	División temporal del dataset y validación estadística	División Train (Ene-Jun 2025: 7.8M) / Validation (Jul-Ago 2025: 2.7M) / Test (Sep-Dic 2025: 5.2M), verificación de estratificación	Datasets finales listos para entrenamiento en formato pickle (comprimido), documento de validación de calidad de datos (verificación de no data leakage, balance de clases por conjunto, estadísticas comparativas train/val/test)
Semana 6	Entrenamiento del modelo Random Forest y optimización de hiperparámetros	GridSearchCV con k-fold=5, entrenamiento con training set (Ene-Jun 2025), evaluación en validation set (Jul-Ago 2025)	Modelo Random Forest entrenado y optimizado, reporte de hiperparámetros seleccionados (n_estimators, max_depth, class_weight), métricas de validación (F1, Recall, Precision) en validation set
Semana 7	Evaluación del modelo en test set temporal y cálculo de métricas finales	Evaluación en test set (Sep-Dic 2025), cálculo de F1-Score, Recall, Precision, AUC-ROC, matrices de confusión, bootstrap (1000 muestras)	**Métricas finales del modelo:** F1-Score, Recall, Precision, AUC-ROC con intervalos de confianza 95 %, curvas ROC, análisis de feature importance, comparación con benchmarks de literatura
Semana 8	Documentación final de instrumentos y preparación de informe	Compilación de todos los entregables anteriores, creación de este documento LaTeX	**Documento final:** Ïnstrumentos para la Constatación del Estado del Problema de Investigación"(PDF, 30-35 páginas), todos los scripts Python documentados (repositorio Git), datasets limpios y validados, modelo serializado

Justificación de Actividades 100 % Cuantitativas

Todas las actividades del cronograma (8 semanas = 2 meses) utilizan **técnicas cuantitativas** exclusivamente:

- **Semana 1-1.5:** Análisis de datos secundarios (extracción y validación de dataset de gestión 2025)
- **Semana 2:** Análisis estadístico descriptivo (EDA con pandas, numpy, matplotlib)
- **Semana 3:** Análisis documental cuantitativo (revisión de metadatos del sistema, NO entrevistas)
- **Semana 4:** Feature engineering (transformaciones numéricas, agregaciones temporales)
- **Semana 5:** Validación estadística del dataset (división temporal 50/17/33, verificación de calidad)
- **Semana 6:** Entrenamiento de modelo Random Forest y optimización de hiperparámetros
- **Semana 7:** Evaluación estadística del modelo en test set temporal (métricas finales)
- **Semana 8:** Documentación técnica y compilación de entregables

NO se realizan técnicas cualitativas:

× Entrevistas formales estructuradas o semiestructuradas

× Grupos focales con stakeholders

× Observación participante o shadowing del equipo de fraude

× Análisis de contenido cualitativo (codificación, categorías emergentes)

× Análisis narrativo o fenomenológico

9.1. Triangulación Metodológica en Investigación Cuantitativa

Según **Hernandez2014<empty citation>**, la triangulación NO es exclusiva de enfoques mixtos o cualitativos. En estudios cuantitativos, la triangulación fortalece la **validez de constructo** mediante la convergencia de múltiples técnicas de medición sobre el mismo fenómeno.

Aplicación de triangulación cuantitativa en esta investigación:

1. **Triangulación temporal de datos:**
   - Training set Ene-Jun 2025 (7.8M transacciones) para entrenamiento del modelo

- Validation set Jul-Ago 2025 (2.7M transacciones) para calibración de hiperparámetros
- Test set Sep-Dic 2025 (5.2M transacciones) para evaluación final independiente
- Validación cruzada temporal: verificar que patrones de fraude detectados en Ene-Jun persisten en Sep-Dic

2. **Triangulación metodológica (técnicas cuantitativas convergentes):**
   - *Técnica 1:* Análisis estadístico descriptivo (EDA) para identificar patrones de fraude
   - *Técnica 2:* Machine Learning supervisado (Random Forest) para clasificar transacciones
   - *Técnica 3:* Análisis estadístico inferencial (intervalos de confianza bootstrap, pruebas de hipótesis) para validar significancia de diferencias entre fraude/no fraude
   - *Convergencia:* Si las tres técnicas identifican las mismas variables como predictoras clave (ej: `amount`, `tx_count_24h`, `gateway`), se fortalece la validez
3. **Triangulación de medición (múltiples indicadores por constructo):**
   - Constructo: "Precisión del modelo de detección de fraude"
   - Indicador 1: F1-Score ($\geq 85\,\%$) - balance entre Precision y Recall
   - Indicador 2: Recall ($\geq 90\,\%$) - capacidad de detectar fraudes reales
   - Indicador 3: Precision ($\geq 80\,\%$) - exactitud de alertas
   - Indicador 4: AUC-ROC (≥ 0.92) - capacidad discriminativa global
   - *Validación:* Si los 4 indicadores se cumplen simultáneamente, se confirma que el modelo es efectivo
4. **Triangulación de investigadores (validación de etiquetas):**
   - Fuente 1: Etiquetas automáticas del sistema (`is_fraud` basado en chargebacks)
   - Fuente 2: Validación con equipo de Contabilidad/Fraude de TechSport (opcional, ver sección 7.1, ítem 2)
   - Acuerdo inter-rater: Cohen's Kappa >0.8 indicaría alta confiabilidad del etiquetado

Conclusión metodológica: Esta investigación **SÍ utiliza triangulación cuantitativa** para fortalecer la validez interna mediante convergencia de múltiples fuentes de datos, técnicas estadísticas, y métricas de evaluación. Esto NO contradice el enfoque exclusivamente cuantitativo, sino que lo refuerza según los principios establecidos por **Hernandez2014<empty citation>**.

10. Referencias Metodológicas

Nota: Las referencias completas en formato APA 7ª edición se encuentran en el archivo `bibliografia/referencias.bib`. A continuación se listan las principales fuentes metodológicas citadas en este documento:

- **Hernández Sampieri et al. (2014):** Metodología de la investigación (6ª ed.). Capítulos sobre enfoque cuantitativo, validez de instrumentos, triangulación metodológica, y análisis de datos secundarios.
- **Hafez et al. (2025):** Revisión sistemática sobre técnicas de IA para detección de fraude en tarjetas de crédito. Benchmark de métricas (F1-Score: 85-94 %).
- **Baesens et al. (2015):** Fraud Analytics Using Descriptive, Predictive, and Social Network Techniques. Técnicas de detección de fraude y validación de modelos.
- **Hernández Aros et al. (2024):** Revisión de literatura sobre detección de fraude financiero mediante Machine Learning.
- **Géron (2022):** Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. Implementación práctica de Random Forest y métricas de evaluación.

Referencias completas citadas: