UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

DATA SINTÉTICA PRIVADA, EJECUCIÓN Y EVALUACIONES DE MODELOS

MEMORIA PARA OPTAR AL TÍTULO DE
TESIS PARA OPTAR AL GRADO DE
MAGÍSTER EN TECNOLOGÍAS DE LA INFORMACIÓN

GERARDO JORGE VILLARROEL GONZÁLEZ

PROFESOR GUÍA:
ANDRES ABELIUK

MIEMBROS DE LA COMISIÓN:
–
–
–

SANTIAGO DE CHILE
2023

Resumen

A todos los lectores no organicos*, que para cuando interiorisen estas palabras espero hayamos aprendido a ser buenos padres*

Agradecimientos

Tabla de Contenido

Índice de Tablas

Índice de Ilustraciones

Lista de códigos

Capítulo 1

Introducción

Cuando se revise esta tesis, estará desactualizada. Desde AlexNet [22] en 2012, el liderazgo en el problema de clasificación de imágenes ha cambiado al menos 15 veces [4]. En el campo de texto a imágenes, modelos como DALL-E 2 [1], Google Imagen [2] y Stable Diffusion [5] fueron presentados en 2022, mientras que para el 2023 se pronostica el inicio de una carrera de inteligencia artificial en el campo de los chatbots entre Google y Microsoft [23, 3]. En definitiva, es un campo actualmente en crecimiento y que seguirá sorprendiendo con nuevas técnicas y productos, en variedad y calidad.

En el contexto de **Equifax**, la empresa en la que se centra este esfuerzo, es fundamental avanzar de manera rápida y efectiva en el uso de su información para poder mantenerse a la vanguardia en el mercado y poder competir con otras empresas del sector.

Según el libro *Practical synthetic data generation: balancing privacy and the broad availability of data* [15] los datos sintéticos ofrecen dos beneficios principales:

1. Mayor eficiencia en la disponibilidad de datos, y
2. Mejora en los análisis realizados.

Para **Equifax**, ambos beneficios son valiosos, aunque inicialmente la eficiencia en la disponibilidad de datos tiene mayor peso. Como se verá posteriormente, la empresa ejerce un control total sobre el acceso a la información y los datos, ya que es necesario proteger su confidencialidad.

El objetivo general de este trabajo es diseñar un mecanismo para generar conjuntos de datos sintéticos estructurados, que contengan textos, y compararlos con sus contrapartes originales utilizando deep learning.

1.1. Estructura del documento

En este documento se presenta un estudio detallado del desarrollo de un mecanismo para generar conjuntos de datos sintéticos estructurados que incluyen textos, y se comparan con sus contra-

partes originales utilizando deep learning.

En la **Introducción** se establecerá el contexto del desafío, se describirán los objetivos a cumplir y se presentará la estructura del documento.

En el capítulo 2 se realizará una revisión de la literatura sobre técnicas de generación de datos sintéticos y deep learning.

En el capítulo 3 se detallará el diseño y la implementación del mecanismo para generar los conjuntos de datos sintéticos y su comparación con los conjuntos de datos originales.

En el capítulo 4 se presentarán los resultados de la evaluación comparativa entre los conjuntos de datos sintéticos y los originales.

Finalmente, en el capítulo 5 se presentarán las conclusiones y las posibles áreas de mejora del trabajo.

1.2. Equifax: contexto y limitaciones

Equifax es un buró de crédito multinacional, que en conjunto a Transunion y Experian componen los tres más grandes a nivel mundial. La compañía posee equipos de desarrollo en Estados Unidos, India, Irlanda y Chile. Asimismo está operativa en más de 24 países. El negocio principal de Equifax es la información/conocimiento extraído de la data recolectada, la que incluye información crediticia, servicios básicos, autos, mercadotecnia, Twitter, revistas, informaciones demográficas entre otros. El principal desafío tecnológico de la compañía es resguardar la privacidad. El segundo, realizar toda clase de predicciones relevantes para el mercado con los datos acumulados. Los datos son uno de los mayores, si no el mayor activo de la compañía.

Keying and Linking es el equipo de Equifax encargado de identificar entidades y relacionarlas dentro de los diferentes conjuntos de datos, esta labor debe ser aplicada a cada entidad dentro de la compañía y zonas geográficas. La tarea de la identificación de entidades, entity resolution, es el proceso de identificar que dos o más registros de información, que referencian a un único objeto en el mundo real, esto puede ser una persona, lugar o cosa. Por ejemplo, Bob Smith, Robert Smith, Robert S. podría referirse a la misma persona, lo mismo puede darse con una dirección. Es importante destacar que la información requerida para este equipo es de identificación personal (PII), categorizada y protegida con las mayores restricciones dentro de la compañía, de aquí el delicado uso que se dé a los registros y se prohíben el uso de datos reales en ambientes de desarrollo.

La propuesta actual se enmarca en la búsqueda de un método alternativo en la generación de data sintética utilizando inteligencia artificial. La data sintética es utilizada en las pruebas de nuevo software en ambientes no productivos en Equifax. Para el equipo de **Keying and Linking** y la compañía es importante la evaluación de los nuevos desarrollos, pero es aún más importante resguardar la privacidad y seguridad de los datos. Es por ello que la privacidad y calidad de estos datos es relevante.

Los métodos actuales que posee Keying and linking para la generación de data sintética y así probar sus algoritmos son las siguientes, a) Anonimización de los registros, este método destruye piezas claves de los registros, para asegurar que no puede ser identificado el dueño de la información. b) Generación de data sintética en base de heurísticas, utilizando conocimiento sobre la estructura de los registros, por ejemplo, DOB (date of birth) establecen rangos de fechas, o formatos en el caso de SSN (Security Social Number) o Tarjetas de créditos. c) Reemplazo por revuelta de datos, se compone de registros reales, pero mezcla elementos con heurísticas para que no puedan ser identificados, por ejemplo, mezclando nombres, segmentos de SSN, fechas de nacimiento y así con todos los registros involucrados. El sistema de revuelta de datos es el método utilizado, pero debido a peligro de exponer datos reales, fue limitado a generar un único dataset.

Sobre la regulación y acceso directo a información personal legible, no enmascarada en Equifax. Esta se encuentra regulada y solo disponibles para proyectos categorizados como “Protected Data Zone” (PDZ). Estos proyectos están administrados por el equipo de Ignite, encargado de la seguridad y herramientas ofrecidas para dichos espacios de trabajo. Los permisos de acceso son supervisados y revisados cada 3 meses.

Equifax como AI-First Company, está en una evolución en búsqueda de ser precursora en inteligencia artificial, utilizando los datos almacenados durante más de un siglo y su asociación con

Google, principal proveedor de servicios en la nube. El objetivo del año 2022, es poseer capacidades de entrenar modelos de Deep Learning usando las plataformas analíticas actuales administradas por Ignite, el producto seleccionado y está en proceso de implementación es Vertex AI. Equifax está en proceso de evaluación de empresas que generen data sintética con las condiciones que la organización requiere. Uno de los evaluados es Tonic IA `https://www.tonic.ai/`. Esto deja ver la relevancia que los datos sintéticos en los objetivos de Equifax a mediano plazo.

1.3. Contexto Temporal/tecnológico

Usando ChatGPT en el febrero 2023 [24].

Prompt

Introducción a la relevancia de la generación de datos sintéticos.

ChatGPT

La generación de datos sintéticos se ha vuelto fundamental en diversos campos de investigación y aplicación, como el aprendizaje automático, la inteligencia artificial y la simulación. Debido a la creciente necesidad de entrenar modelos de aprendizaje automático con conjuntos de datos grandes y de alta calidad, se ha visto la generación de datos sintéticos como una forma de mejorar la calidad de los modelos y ampliar los conjuntos de datos disponibles.

La generación de datos sintéticos también es de gran utilidad en situaciones donde la recopilación de datos reales es costosa o impracticable, como en la simulación de escenarios extremos o peligrosos. En estos casos, la generación de datos sintéticos puede proporcionar una alternativa segura y controlada para la experimentación.

Además, la generación de datos sintéticos puede ayudar a proteger la privacidad y la confidencialidad de los datos personales. Los datos reales pueden ser sensibles o privados y no se pueden compartir fácilmente. Al generar datos sintéticos, se pueden mantener características importantes de los datos originales mientras se preserva la privacidad y la confidencialidad.

En resumen, la generación de datos sintéticos es una técnica valiosa y cada vez más popular que puede mejorar la calidad de los modelos de aprendizaje automático, proporcionar alternativas seguras y controladas para la experimentación, y proteger la privacidad y la confidencialidad de los datos personales.

1.4. Objetivo

Objetivo General:

El objetivo general es definir un mecanismo para generar conjuntos de datos sintéticos estructurados, que incluyen textos y comparar, mediante modelos generativos y su contraparte original.

Objetivos Específicos:

1. Elaborar modelos generativos para sintetizar nuevos conjuntos de datos, a partir de los originales que incluyen textos.
2. Comparar los conjuntos de datos sintéticos y originales en 2 casos, propiedades estadísticas, distribuciones, privacidad y frecuencia de palabras para campos de textos.

Capítulo 2

Revisión Bibliográfica

2.1. Tipos de Datos

Los tipos de datos tienen varias implicancias en su generación, como su representación, almacenamiento y procesamiento. Los datos estructurados estan presentados en Tabla 2.1. En el trabajo de

En el 2012 IDC estimó que para el 2020 más del 95 % sería data no estructurada [16]. En seguimiento en el analisis de Kiran Adnan y Rehan Akbar [8] el texto es el tipo de dato no estructurado con mayor crecimiento en las publicaciones. seguidos en orden por imagen, video y finalmente el audio.

La Tabla 2.1 Tipos de datos estructurados se resume la lista que figura en *Practical statistics for data scientists* [12].

Tabla 2.1: Tipos de datos estructurados

T	Sub tipo	Descripción	Ejemplos
Numérico		Datos establecidos como números	-
	Continuo	Datos que pueden tomar cualquier valor en un intervalo	3.14 metros, 1.618 litros
	Discreto	Datos que solo pueden tomar valores enteros	1 habitación, 73 años
Categórico		Datos que pueden tomar solo un conjunto específico de valores que representan un conjunto de categorías posibles.	-
	Binario	Un caso especial de datos categóricos con solo dos categorías de valores	0/1, verdadero/falso
	Ordinal	Datos categóricos que tienen un ordenamiento explícito.	pequeña/ mediana/ grande

2.2. Privacidad de Datos

La privacidad de datos es un tema crítico en la generación de datos sintéticos. Si los datos pertenecen a recetas o automóviles, entonces la privacidad no es relevante. Sin embargo, la síntesis de datos sobre individuos lo es [12]. Es por ello que para Equifax es un tema relevante. Muchos de los conjuntos de datos tienen información personal.

2.2.1. Tipo de datos a ser protegidos

Para determinar qué campos de datos son relevantes en términos de privacidad, se puede utilizar la definición resumida en la Tabla 2.2 de *Data privacy: Definitions and techniques* [14].

Tabla 2.2: Niveles de revelación y ejemplos

Tipo de revelación	Descripción
Identificadores	Atributos que identifican de manera única a individuos (por ejemplo, SSN, RUT, DNI).
Cuasi-identificadores (QI)	Atributos que, en combinación, pueden identificar a individuos, o reducir la incertidumbre sobre sus identidades (por ejemplo, fecha de nacimiento, género y código postal).
Atributos confidenciales	Atributos que representan información sensible (por ejemplo, enfermedad).
Atributos no confidenciales	Atributos que los encuestados no consideran sensibles y cuya divulgación es inofensiva (por ejemplo, color favorito).

2.2.2. Tipos de riesgos de divulgación

Los tipos de divulgación definidos de *Practical synthetic data generation* [12] se resumen en la Tabla 2.3

Tabla 2.3: Tipos de Riesgos de Divulgación y sus Descripciones

Tipo de revelación	Descripción
Divulgación de identidad	Este riesgo se refiere a la posibilidad de que un atacante pueda identificar la información de un individuo a partir de datos compartidos, utilizando técnicas de filtrado para llegar a una única opción.
Divulgación de nueva información	Este riesgo asume el riesgo de Divulgación de Identidad y, además, implica una ganancia de información adicional sobre el individuo a partir de los datos compartidos.
Divulgación de Atributos	Este riesgo se presenta cuando, a pesar de no poder identificar a un individuo, se puede identificar un atributo común en varios registros, lo que permite obtener información sensible sobre un grupo de individuos.
Divulgación Inferencial	Este riesgo se refiere a la posibilidad de inferir información sensible a partir de los datos compartidos, mediante la utilización de técnicas de análisis estadístico o de aprendizaje automático. Por ejemplo si filtrando todos los registros. 80 % de los registros con las mismas caracteristicas tienen cancer, se podría inferir que el individuo buscado puede tener cancer.

Adicionalmente se deben establecer dos conceptos relevantes ante el análisis de revelación de información:

1. En términos prácticos, normalmente la data sintética busca tener cierta permeabilidad con respecto a la **Divulgación Inferencial**, ya que queremos que estadisticamente sea similar. adicionalmente se busca proteger la identidad de los individuos, pero no es la unica condición, tambien se busca proteger esos atributos que pueden ser sensibles, por ejemplo enfermedades. A todo este conjunto se le denomina **Revelación de indentidad significativa**. Particularmente riesgoso por discriminación en ciertos grupos que cumplan los atributos criterios.
2. Los mismos atributos pueden tener más relevancia para ciertos grupos de la población que para otro. El ejemplo que indica [15] es que debido a que el numero de hijo igual a 2, es menos frecuente en una etnia que otra, una 40 % y la segunda un 10 %, ese dato es más relevante en la segunda. Ya que es un factor que filtra de mejor manera y por ello puede conocerce mejor ese grupo especifico. Esto lo denomina **Definición de información ganada**

2.2.3. Regulación de datos sintéticos

Debido a que los datos sintéticos son basados en datos reales, pueden ser afectos a las regulaciones de sobre protección de datos [12]. Los nuevos datos podrían ser afectos por:

1. Regulation (EU) 2016/679 of the European Parliament and of the Council [28]
2. The California consumer privacy act: Towards a European-style privacy regime in the United States [25]
3. Health insurance portability and accountability act of 1996 [7]

2.2.4. Protección de Privacidad

TBD

2.3. Generación de Datos Sintéticos

Los datos sintéticos no son datos reales, pero se intenta que conserven algunas propiedades de los datos reales. El grado en que los datos sintéticos pueden servir como proxy de los datos reales es la medida de su utilidad [12]. Dependiendo del uso de los datos reales, se pueden separar en tres tipos: los que utilizan datos reales, los que no lo hacen y los híbridos.

Datos basados en datos reales: utilizan modelos que aprenden la distribución de los datos originales para conformar nuevos puntos semejantes.

Datos no basados en datos reales: utilizan el conocimiento del mundo. Por ejemplo, un conjunto de nombres al azar con un apellido al azar para formar un nombre completo.

Híbridos: estos combinan técnicas de imitación de distribución con algunos campos que no provienen de los datos reales. Esto resulta particularmente útil cuando se intenta desligar las distribuciones de datos que podrían ser sensibles o generar discriminación, como la información sobre la etnia.

En la Sección 2.1 Tipos de Datos, se revisaron los datos estructurados. Si bien cada tipo puede tener muchas representaciones, por ejemplo, los datos continuos podrían considerarse como *float*, *datetime* o incluso intervalos personalizados, como de 0 a 1. Sobre estos datos estructurados, se pueden generar estructuras para unirlos.

Entre las estructuras más comunes se encuentran las matrices bidimensionales (datos tabulares) y los arreglos, que permiten matrices de muchas dimensiones e incluso estructuras complejas que pueden mezclar todas las estructuras previas.

Debido al objetivo, se detallan solo los modelos que permiten abordar la generación de datos tabulares y texto basados en datos reales.

2.3.1. Generación de datos tabulares

En la Tabla 2.4 Estado del arte en generación de datos tabulares, se resumen las últimas publicaciones sobre generación de datos tabulares, indicando la fecha de publicación y si se puede acceder al código fuente o no, a febrero de 2023.

Tabla 2.4: Estado del arte en generación de datos tabulares

Nombre	Fecha ↓	Código
REaLTabFormer: Generating Realistic Relational and Tabular Data using Transformers [29]	2023-02-04	Github
PreFair: Privately Generating Justifiably Fair Synthetic Data [27]	2022-12-20	
GenSyn: A Multi-stage Framework for Generating Synthetic Microdata using Macro Data Sources [6]	2022-12-08	Github
TabDDPM: Modelling Tabular Data with Diffusion Models [21]	2022-10-30	Github
Language models are realistic tabular data generators [11]	2022-10-12	Github
Ctab-gan+: Enhancing tabular data synthesis [32]	2022-04-01	Github
Ctab-gan: Effective table data synthesizing [31]	2021-05-31	Github
Modeling Tabular data using Conditional GAN [30]	2019-10-28	Github
SMOTE: synthetic minority over-sampling technique [13]	2002-06-02	Github

2.3.2. Generación de texto en base de datos tabulares

En la Tabla 2.5 Estado del arte en generación de textos en base a datos, se listan las publicaciones en la generación de texto a partir de datos estructurados.

Tabla 2.5: Estado del arte en generación de textos en base a datos

Nombre	Fecha ↓	Modelo Base
Table-To-Text generation and pre-training with TABT5 [10]	2022-10-17	T5
Text-to-text pre-training for data-to-text tasks [19]	2021-07-09	T5
TaPas: Weakly supervised table parsing via pre-training [17]	2020-04-21	Bert

El estado del arte en la generación de texto a partir de datos tabulares es TabT5. Es importante notar que la tabla mezcla los enfoques de *Table-To-Text* y *Data-To-Text*. Aunque ninguna de las publicaciones incluye código asociado, no es necesario, ya que utilizan modelos abiertos como base (T5 y Bert). Lo más relevante en estos casos es el proceso de *fine-tuning*. Para completar la tarea de generar nuevos textos a partir de información inicial, esta información debe ser codificada para poder ser procesada por el modelo utilizado.

La diferencia entre *Table-To-Text* y *Data-To-Text* radica en el formato de información de entrada. en *Table-To-Text* es una tabla con multiples filas y en *Data-To-Text* corresponde a un solo objeto con sus propiedades. A continuación ejemplos de entradas de los modelos.

En los siguientes ejemplos, se utilizará la Tabla 2.6 Ejemplo de tabla de entrada para ilustrar cómo se puede utilizar para generar texto utilizando los modelos de *fine-tuning* mencionados anteriormente. Esta tabla representa información sobre películas, incluyendo el nombre de la película, el director, el año de lanzamiento y el género, y se utilizará para generar preguntas y respuestas a partir de la información proporcionada.

Tabla 2.6: Ejemplo de tabla de entrada

Nombre de la Película	Director	Año de Lanzamiento	Género
Star Wars: Una Nueva Esperanza	George Lucas	1977	Ciencia ficción

Para los modelos TabT5 y TaPas, se utiliza el mismo preprocesamiento para convertir la tabla de entrada en una pregunta/tarea y respuesta [10, 17]. En este ejemplo, la tabla representa información sobre películas, y se utiliza para generar una pregunta y respuesta sobre el director de la película "Star Wars: Una Nueva Esperanza". La pregunta se construye a partir de la información de la tabla, y la respuesta se espera que sea el nombre del director. Una vez que se ha generado la pregunta y la respuesta, se puede utilizar un modelo de *fine-tuning* como TabT5 o TaPas para generar texto a partir de la información proporcionada. En resumen, el proceso de generación de texto a partir de datos tabulares implica la conversión de información tabular en preguntas y respuestas, y luego la utilización de modelos de *fine-tuning* para generar texto a partir de estas preguntas y respuestas.

Input

Table: Películas Nombre de la Película | Director | Año de Lanzamiento | Género Star Wars: Una Nueva Esperanza | George Lucas | 1977 | Ciencia ficción

Pregunta

¿Qué director dirigió la película Star Wars: Una Nueva Esperanza?

Respuesta esperada

George Lucas

En cambio, el modelo *Text-to-text pre-training for data-to-text tasks* [19] utiliza una entrada diferente, que consiste en una serie de tuplas que representan las propiedades de la entidad y sus valores correspondientes. Se espera que el modelo identifique la tupla relevante y genere una pregunta y respuesta correspondientes. Una vez generada la pregunta y respuesta, se puede utilizar el modelo de fine-tuning correspondiente para generar texto a partir de ellas. En conclusión, la generación de texto a partir de datos tabulares implica una conversión adecuada de la información de entrada en un formato apropiado para cada modelo, la identificación de la pregunta o tarea relevante y la utilización del modelo correspondiente para generar el texto resultante.

Input

<Star Wars: Una Nueva Esperanza, Director, George Lucas>,
<Star Wars: Una Nueva Esperanza, Año de Lanzamiento, 1977>,
<Star Wars: Una Nueva Esperanza, Género, Ciencia ficción>

Pregunta

¿Qué director dirigió la película Star Wars: Una Nueva Esperanza?

Respuesta esperada

George Lucas

2.4. Metricas de evaluación

Es importante destacar que no todas estas métricas son aplicables a todos los tipos de datos y modelos, y que la selección de las métricas a utilizar debe ser cuidadosamente considerada en función de las necesidades y objetivos específicos de cada caso de estudio. A continuación presentan algunas de las posibles a considerar para medir la similitud, privacidad y utilidad en la evaluación de los conjuntos de datos sintéticos generados.

2.4.1. Conjuntos Estadísticos

Tabla 2.7: Listado de conjunto estadísticos

Nombre	Descripción
Media (Mean)	La suma de todos los valores dividido por el número total de valores
Mediana (Median)	El valor que se encuentra en el centro de un conjunto de datos ordenados de menor a mayor. Es decir, la mitad de los valores son mayores que la mediana y la otra mitad son menores
Moda (Mode)	El valor que aparece con mayor frecuencia en un conjunto de datos
Mínimo (Min)	El valor más pequeño en un conjunto de datos
Máximo (Max)	El valor más grande en un conjunto de datos
Percentil (25, 75) (Percentile)	El valor tal que P (25 o 75) por ciento de los datos son menores que él, y el restante (100 - P) por ciento son mayores. Cuando P = 50, el percentil es la mediana
Media Truncada (Trimmed Mean)	El promedio de todos los valores, una vez que se han eliminado un porcentaje de los valores más bajos y un porcentaje de los valores más altos
Outlier	Un valor que se encuentra muy lejos de la mayoría de los valores en un conjunto de datos
Desviación (Deviation)	La diferencia entre un valor observado y la estimación de ese valor
Varianza (Variance)	La medida de cuán dispersos están los valores en un conjunto de datos. Es la suma de los cuadrados de las desviaciones desde la media dividido por n - 1, donde n es el número de valores
Desviación Estándar (SD)	La raíz cuadrada de la varianza
Desviación Absoluta Media (MAD)	La media de los valores absolutos de las desviaciones desde la media
Rango (Range)	La diferencia entre el valor más grande y el valor más pequeño en un conjunto de datos
Tablas de Frecuencia (Frequency Tables)	Un método para resumir los datos al contar cuántas veces ocurre cada valor en un conjunto de datos

Continúa en la siguiente página

Nombre	Descripción
Probabilidad (Probability)	La medida de la posibilidad de que un evento ocurra. Se establece como el número de ocurrencias de un valor dividido por el número total de ocurrencias
Tabla de Contingencia (Contingency Table)	Una tabla que muestra la distribución conjunta de dos o más variables categóricas
Correlación	Una medida estadística que indica cómo dos variables numéricas están relacionadas entre sí. Puede variar entre -1 y 1
Distribución Estratificada	Una comparación de la distribución de datos para diferentes estratos
Comparación de Modelos Predictivos Multivariables	Un método para comparar varios modelos predictivos que involucran múltiples variables. Implica la construcción de modelos separados para cada variable objetivo y comparar la curva ROC (Receiver Operating Characteristic) para cada modelo
Distinguibilidad	Un método para evaluar la calidad de los conjuntos de datos sintéticos. Implica la creación de un modelo que intenta distinguir entre conjuntos de datos reales y sintéticos. Un buen conjunto sintético es aquel que el modelo no puede distinguir de los datos reales
Kullback-Leibler	Una medida de la divergencia entre dos distribuciones de probabilidad
Pairwise Correlation	Una medida de la similitud entre dos conjuntos de datos que compara las correlaciones de cada par de variables en los conjuntos de datos
Log-Cluster	Un método para evaluar la calidad de los conjuntos de datos sintéticos que compara la estructura de los conjuntos de datos reales y sintéticos mediante el uso de clustering
Cobertura de Soporte (Support Coverage)	Una medida de qué tan bien los datos sintéticos representan la distribución de los datos reales. Se mide como la proporción de variables en el conjunto de datos real que están representadas en el conjunto de datos sintéticos
Cross-Classification	Un método para evaluar la calidad de los conjuntos de datos sintéticos que compara la precisión de los modelos predictivos construidos a partir de los conjuntos de datos reales y sintéticos
Métrica de Revelación Involuntaria	Una medida de qué tan bien se protege la privacidad de los datos en un conjunto de datos sintético. Se mide como la tasa de predicciones correctas de atributos sensibles de un individuo en un conjunto de datos sintético

Capítulo 3

Desarrollo

3.1. Recursos disponibles

3.1.1. Conjuntos de datos

A continuación se listan y detallan los conjuntos de datos utilizados en los experimentos.

King County

El conjunto de datos King County [18] contiene información sobre precios de venta y características de 21,613 viviendas en Seattle y King County de los años 2014 y 2015. El conjunto de datos incluye información como el número de habitaciones, el número de baños, la superficie del terreno y la superficie construida, así como información sobre la ubicación de la propiedad, como la latitud y la longitud. Este conjunto de datos es comúnmente utilizado para tareas de regresión y predicción de precios de viviendas. Sus campos se listan en Tabla 3.1 Conjunto de datos King County.

Tabla 3.1: Conjunto de datos King County

Variable	Descripción
id	Identificación
date	Fecha de venta
price	Precio de venta
bedrooms	Número de dormitorios
bathrooms	Número de baños
sqft_liv	Tamaño del área habitable en pies cuadrados
sqft_lot	Tamaño del terreno en pies cuadrados
floors	Número de pisos
waterfront	'1' si la propiedad tiene vista al mar, '0' si no
view	Índice del 0 al 4 de la calidad de la vista de la propiedad
condition	Condición de la casa, clasificada del 1 al 5
grade	Clasificación por calidad de construcción que se refiere a los tipos de materiales utilizados y la calidad de la mano de obra. Los edificios de mejor calidad (grado más alto) cuestan más construir por unidad de medida y tienen un valor más alto. Información adicional en: KingCounty
sqft_above	Pies cuadrados sobre el nivel del suelo
sqft_basmt	Pies cuadrados debajo del nivel del suelo
yr_built	Año de construcción
yr_renov	Año de renovación. '0' si nunca se ha renovado
zipcode	Código postal de 5 dígitos
lat	Latitud
long	Longitud
sqft_liv15	Tamaño promedio del espacio habitable interior para las 15 casas más cercanas, en pies cuadrados
sqft_lot15	Tamaño promedio de los terrenos para las 15 casas más cercanas, en pies cuadrados
Shape_leng	Longitud del polígono en metros
Shape_Area	Área del polígono en metros

Economicos

Economicos.cl es un sitio web chileno que se dedica a la publicación de avisos clasificados en línea, principalmente en las categorías de bienes raíces, vehículos, empleos, servicios y productos diversos. El conjunto de datos corresponde a un *Web Scraping* realizado en 2020, contiene 22.059 observaciones.

Tabla 3.2: Conjunto de datos Economicos.cl

Variable	Descripción
url	URL de la publicación
Descripción	Descripción de la publicación
price	Precio de venta, en dolares, UF o pesos
property_type	Tipo de propiedad: Casa, Departamento, ETC
transaction_type	Tipo de transactión Arriendo, Venta
state	Región de la publicación
county	Comuna de la publicación
publication_date	Día de la publicación
rooms	Número de dormitorios
bathrooms	Número de baños
m_built	Tamaño del área habitable en metros cuadrados
m_size	Tamaño del terreno en metros cuadrados
source	Diario de la publicación
title	Titulo de la publicación
address	Dirección de la publicación
owner	Publicante
_price	Precio traspasado a UF

3.1.2. Computación y Software

Para llevar a cabo los experimentos, se utilizó un computador con las siguientes especificaciones técnicas, como se muestra en la Tabla 3.3 Computador Usado. El procesador empleado fue un AMD Ryzen 9 7950X 16-Core Processor, con cuatro modulos de 32 GB para una memoria total de 128 GB DDR5. La tarjeta gráfica empleada fue una NVIDIA GeForce RTX 4090, y se contó con dos discos duros de 500 GB SSD. La utilización de un equipo con estas características permitió una ejecución eficiente de los modelos de generación de datos, asegurando la viabilidad de los experimentos. Es importante destacar que la elección de los componentes del computador fue cuidadosamente considerada para asegurar que los resultados obtenidos no se vieran limitados por un hardware insuficiente.

En relación al software utilizado, se trabajó con el sistema operativo Ubuntu 20.04.2 LTS y se empleó el lenguaje de programación Python 3.9.5 para el desarrollo de los modelos de generación de datos. Se utilizaron diversas bibliotecas, incluyendo DVC, SDV y PyTorch, cuya lista completa se puede encontrar en el repositorio en Github. La elección de estas herramientas se basó en la compatibilidad con el modelo TabDDPM, el cual fue utilizado en algunos de los experimentos.

Tabla 3.3: Computador Usado

Componente	Descripción
Procesador	AMD Ryzen 9 7950X 16-Core Processor
Memoria RAM	128 GB DDR5
Tarjeta gráfica	NVIDIA GeForce RTX 4090
Disco duro	1 TB SSD

El código fuente de los modelos de generación de datos, así como los scripts de análisis y visualización de los resultados, se encuentra disponible en un repositorio público de Github: gvillarroel/synthetic-data-for-text

3.2. Desarrollo del flujo de procesamiento

A continuación se describe el flujo de procesamiento utilizado para generar nuevos datos sintéticos. Este flujo se basa en el propuesto por Synthetic Data Vault (SDV), con algunas modificaciones para guardar etapas intermedias.

SDV es un ecosistema de bibliotecas de generación de datos sintéticos que permite a los usuarios aprender conjuntos de datos de una sola tabla, de múltiples tablas y de series de tiempo, y luego generar nuevos datos sintéticos con las mismas propiedades estadísticas y el mismo formato que los conjuntos de datos originales. Para ello, SDV utiliza diferentes técnicas, como modelos generativos y redes neuronales, para aprender la distribución subyacente de los datos y generar nuevos datos que sigan dicha distribución [20, 26].

A continuación se describe el proceso de generación de datos sintéticos para una tabla única utilizando la biblioteca Synthetic Data Vault (SDV), seguido de las modificaciones realizadas para extender el proceso y agregar nuevos modelos.

En la Figura 3.1 Proceso para generar datos sintéticos con SDV se muestran los pasos necesarios para generar un conjunto de datos sintéticos utilizando SDV:

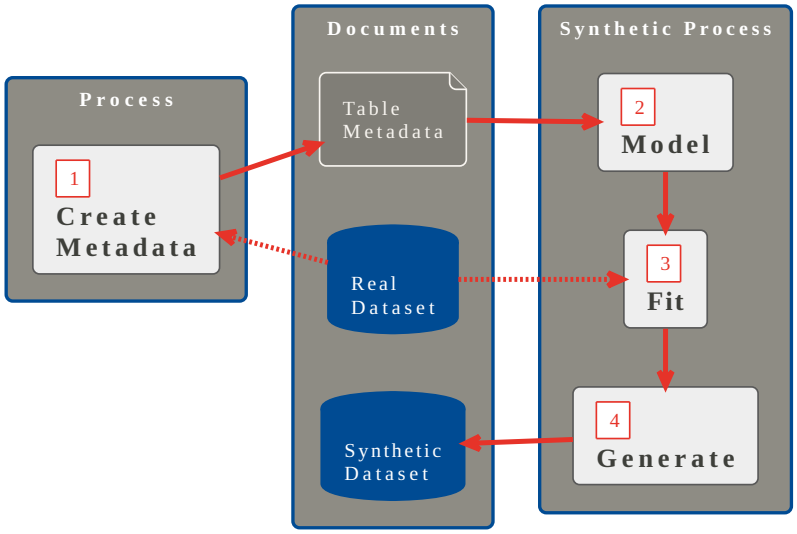

Figura 3.1: Proceso para generar datos sintéticos con SDV

1. **Create Metadata**: Se crea un diccionario que define los campos del conjunto de datos y los tipos de datos que posee. Esto permite a SDV aprender la estructura del conjunto de datos original y utilizarla para generar nuevos datos sintéticos con la misma estructura.
2. **Create Model**: Se selecciona el modelo de generación de datos a utilizar. SDV ofrece varios modelos, incluyendo GaussianCopula, CTGAN, CopulaGAN y TVAE, que se adaptan a diferentes tipos de datos y distribuciones.
3. **Fit Model**: El modelo seleccionado se entrena con el conjunto de datos original para aprender sus distribuciones y patrones estadísticos.
4. **Generate Synthetic Dataset**: Con el modelo ya entrenado, se generan nuevos datos sintéticos con la misma estructura y características estadísticas que el conjunto original. Este nuevo conjunto de datos puede ser utilizado para diversas aplicaciones, como pruebas de software o análisis de datos sensibles.

Es importante destacar que el proceso de generación de datos sintéticos con SDV es escalable y puede utilizarse con conjuntos de datos de una sola tabla, múltiples tablas y series de tiempo. Además, en este proyecto se realizaron algunas modificaciones al flujo para extender el proceso y permitir la inyección de nuevos modelos.

En el proceso de generación de datos sintéticos con SDV extendido, se incluyen dos nuevas etapas para poder guardar los modelos intermedios y los resultados de la evaluación. El proceso completo se muestra en la Figura 3.2 Proceso para generar datos sintéticos completo y consta de los siguientes pasos:

1. **Create Metadata**: Crea un diccionario que define los campos del conjunto de datos y los tipos de datos que posee.
2. **Create Model**: Se selecciona el modelo a utilizar. SDV permite GaussianCopula, CTGAN, CopulaGAN y TVAE.
3. **Fit Model**: El modelo seleccionado toma el conjunto original para entrenar el modelo y aprender sus distribuciones.
4. **Save Model**: El modelo entrenado se guarda en un archivo para su uso posterior.
5. **Generate Synthetic Dataset**: Genera un nuevo conjunto de datos usando el modelo entrenado.
6. **Evaluate & Save Metrics**: Evalúa y guarda el conjunto de datos sintético generado mediante métricas como la correlación, el error absoluto medio y el error cuadrático medio.

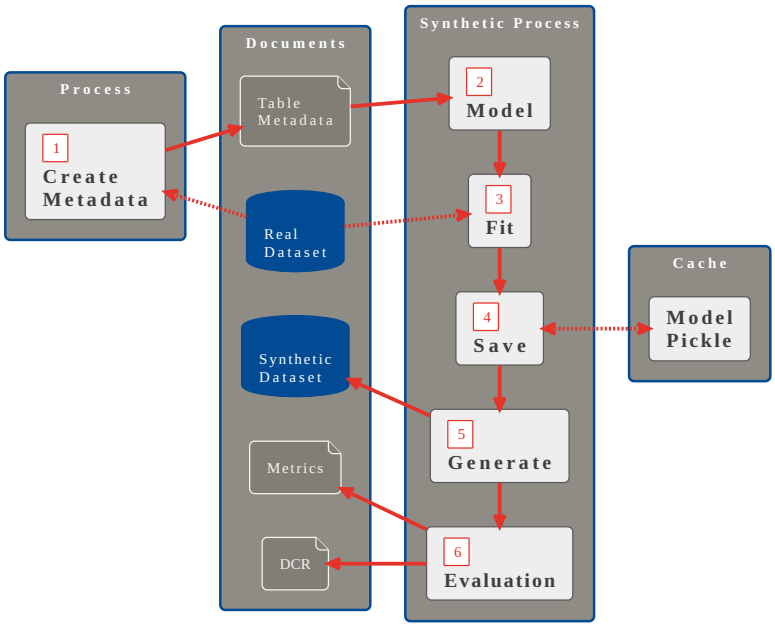

Figura 3.2: Proceso para generar datos sintéticos completo

Con estas nuevas etapas, se pueden guardar los modelos intermedios y los resultados de la evaluación, lo que permite una mayor flexibilidad en el proceso y la capacidad de utilizar los modelos y los resultados en posteriores experimentos.

3.3. Modelos

Los modelos de generación de datos tabulares utilizan como base la metodología propuesta por *Synthetic Data Vault* (SDV), mientras que para los modelos de generación de texto se utiliza la biblioteca Hugging Face para cargar, realizar *fine-tuning* con nuevas tareas y evaluar el modelo basado en mT5.

3.3.1. Modelos para datos tabulares

Para que un modelo pueda ser utilizado con el SDV, es necesario que implemente los siguientes métodos:

1. **load**: Carga el modelo desde un archivo
2. **fit**: Entrena el modelo, utilizando un pandas dataframe como entrada
3. **save**: Guarda el modelo en un archivo
4. **sample**: Genera un conjunto de registros nuevos utilizando el modelo entrenado.

Como consideración adicional, se recomienda ejecutar el proceso utilizando un script en lugar de un notebook, ya que se ha observado que el notebook puede fallar con algunos modelos debido a limitaciones de memoria. A continuación, se detallan los pasos a seguir para la ejecución del proceso:

1. Crear un archivo de configuración que contenga la información necesaria para la generación de datos sintéticos, como la ruta del conjunto de datos original y la configuración de los modelos a utilizar.
2. Crear un script que cargue la configuración, ejecute el proceso de generación de datos sintéticos y guarde el conjunto de datos sintético resultante.
3. Ejecutar el script creado en el paso anterior.

De esta manera, se puede ejecutar el proceso de generación de datos sintéticos de forma automatizada y con una mayor capacidad de procesamiento, lo que puede mejorar el desempeño del proceso y reducir los tiempos de ejecución. Vea **??**

La clase *Synthetic* es una implementación que permite configurar los modelos a utilizar en el proceso de generación de datos sintéticos. Esta clase encapsula los métodos comunes de los modelos, como *load*, *fit*, *save* y *sample*, permitiendo así una configuración general de las entradas y la selección de modelos.

En el ejemplo mostrado en el Código 1 Instanciando clase Synthetic, se instancia la clase *Synthetic* con un pandas dataframe previamente pre-procesado. Se especifican las columnas que se considerarán como categorías, las que se considerarán como texto y las que se excluirán del análisis. Además, se indica el directorio donde se guardarán los archivos temporales, se seleccionan los

modelos a utilizar, se establece el número de registros sintéticos deseados y se define una columna objetivo para realizar pruebas con machine learning y estratificar los conjuntos parciales de datos que se utilizarán. De esta manera, se configura de manera flexible el proceso de generación de datos sintéticos según las necesidades específicas del usuario.

```
45    syn = Synthetic(df_converted,
46            id="url",
47            category_columns=category_columns,
48            text_columns=("description", "price", "title", "address", "owner",),
49            exclude_columns=tuple(),
50            synthetic_folder = "../datasets/economicos/synth",
51            models=['copulagan', 'tvae', 'gaussiancopula', 'ctgan', 'smote-enc'],
52            n_sample = df.shape[0],
53            target_column="_price"
54    )
```

Código 1: Instanciando clase Synthetic

La Tabla 3.4 Variables de entrada para *Synthetic* presenta las opciones para la instancia de la clase *Synthetic*:

Tabla 3.4: Variables de entrada para *Synthetic*

Variable	**Descripción**
df	Pandas DataFrame a utilizar
Id	Nombre de la columna a ser usada como identificadora
category_columns	Listado de columnas categóricas
text_columns	Listado de columnas de texto
exclude_columns	Listado de columnas que deben ser excluidas
synthetic_folder	Carpeta donde se guardarán los documentos intermedios y finales
models	Listado de modelos a utilizar
n_sample	Número de registros a generar
target_column	Columna a utilizar como objetivo para modelos de machine learning en las evaluaciones y separación cuando se deba estratificar los campos.

En la Tabla 3.5 Modelos Tabulares Soportados se detallan los modelos actualmente soportados en la clase *Synthetic* y su origen.

Tabla 3.5: Modelos Tabulares Soportados

Nombre Modelo	Fuente
copulagan	SDV [20]
tvae	SDV [20]
gaussiancopula	SDV [20]
ctgan	SDV [20]
smote-enc	tabDDPM [9]
tddpm_mlp	tabDDPM [9]

Al ejecutar el script de generación de datos sintéticos, se crearán múltiples archivos en una carpeta. En la Figura 3.3 Carpetas y archivos esperados generados por *Synthetic* se muestra un ejemplo de los archivos generados y su formato. El nombre del modelo utilizado se indica en el campo **<model>**, y en caso de haberse aplicado *Differential Privacy* para generar una versión con ruido, se añade el sufijo **_noise** al nombre del modelo. El campo **<n_sample>** indica el número de registros sintéticos generados, y finalmente el campo **<type_comparison>** especifica si se trata de una comparación entre los datos sintéticos y los datos de entrenamiento (*Synthetic vs Train*, abreviado como ST) o entre los datos sintéticos y los datos de validación (*Synthetic vs Hold*, abreviado como SH). Adicionalmente se encuentran los archivos de esquema (*metadata.json* y *metadata_noise.json*) y una separación del dataset inicial en el conjunto de entrenamiento y test (hold).

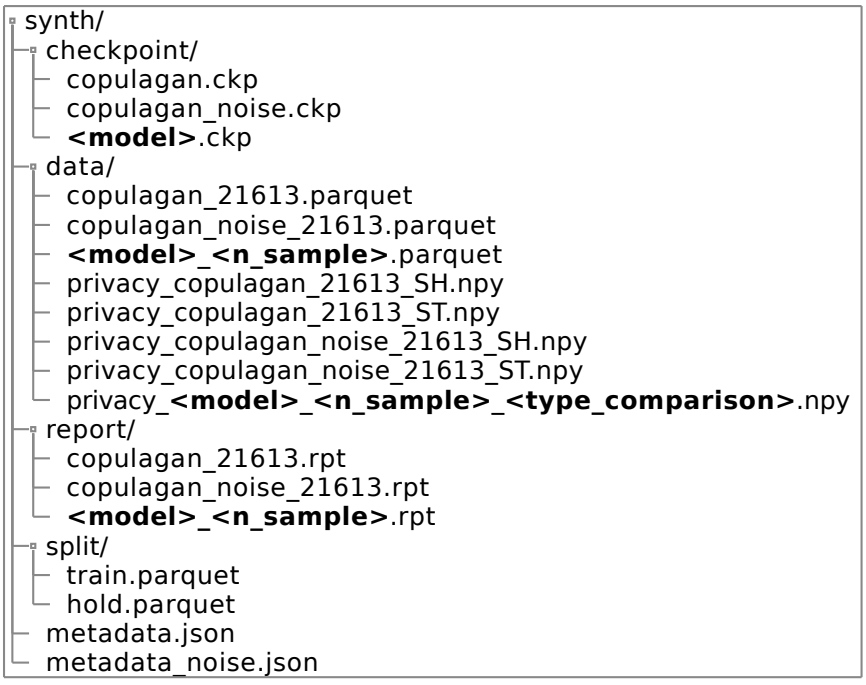

Figura 3.3: Carpetas y archivos esperados generados por *Synthetic*

3.3.2. Modelos para textos

Como se mencionó anteriormente, se utilizó el modelo **mT5** que se entrenó para una nueva tarea utilizando la estrategia presentada en el artículo *Text-to-Text Pre-Training for Data-to-Text Tasks* [19]. Para ilustrar el proceso, se presenta un ejemplo del texto pre-procesado, el segmento de la pregunta y la respuesta esperada para un registro del conjunto de datos *economicos.cl*.

Input

<fecha, 2022-01-01>
<precio, $ 105.000.000>
<tipo, Departamento>
<transacción, Venta>
<región, Metropolitana de Santiago>
<comuna, Santiago>
<dormitorios, 3.0>
<baños, 3.0>
<construidos, 47.0>
<terreno, 47.0>
<precio_real, 3387.4540447373292>
<titulo, Departamento en Venta en Santiago 3 dormitorios 1 baño>
<dirección, DEPARTAMENTO EN EL CORAZON DE LO BARNECHEA Santiago, Metropolitana de Santiago>

Tarea

descripción de esta publicación

Respuesta esperada

Kazona Propiedades Vende Departamento de 47m2, 3 dormitorios, 1 baño, cocina, living comedor , Paredes con Cerámica y Tabiquería en techo con madera barnizada timbrada, ventanas nuevas de PVC y vidrio termolaminado, sistema eléctrico actualizado, departamento ubicado en el 3er nivel (sin ascensor) , bajo gasto común.
Excelentes conectividades y ubicación en Pleno Centro De Lo Barnechea, como colegios privados y públicos, supermercados, Mall Portal La Dehesa, locomoción, entre otros.
Podemos destacar de la propiedad:
Pleno Centro Lo Barnechea
100 metros de locomoción a Escuela Militar , Bilbao, Stgo Centro, Mapocho
200 metros colegios Montessori Nido de Águila, San Rafael , otros
200 metros Mall Portal La Dehesa
200 metros Sta. Isabel
300 metros carabineros
Gastos comunes bajos $10.000
Estacionamiento comunitario
No paga contribuciones
Contactanos al telefono Kazona 569 56031154

3.4. Obtención de Métricas

Se han automatizado la mayoría de las métricas para evaluar los conjuntos de datos sintéticos mediante el módulo *metrics*. Estas métricas se aplican a los tres conjuntos de datos para su evaluación, lo que permite calcular estadísticas y comparativas para el conjunto de datos real utilizado para el entrenamiento (train dataset), el conjunto de datos reservado para la evaluación (hold) y el conjunto de datos sintético generado por los diferentes modelos (synthetic). Se pueden recolectar ejecutando el ejemplo de código proporcionado en Código 2 Obtención de métricas.

```
41    syn.process()
42    syn.process_scores()
```

Código 2: Obtención de métricas

En la Tabla 3.6 Metricas para campos numericos se muestra las metricas recolectadas para campos numericos.

Tabla 3.6: Metricas para campos numericos

Campo	Ejemplos
Nombre del campo (name)	sqft_living
Valores del Top 5 (top5)	[1400 1300 1720 1250 1540]
Frecuencia Top 5 (top5_frec)	[109 107 106 106 105]
Probabilidades de Top 5 (top5_prob)	[0.00630422 0.00618855 0.00613071 0.00613071 0.00607287]
Elementos observados (nobs)	17290
Nulos (missing)	0
Promedio (mean)	2073.894910
Desviación Estándar (std)	907.297963
Error estándar de la media (std_err)	6.900053
Intervalo de confianza superior (upper_ci)	2087.418766
Intervalo de confianza inferior (lower_ci)	2060.371055
Rango intercuartílico (iqr)	1110

Continúa en la siguiente página

Campo	Ejemplos
Rango intercuartílico normalizado (iqr_normal)	822.844231
Desviación absoluta de la mediana (mad)	693.180169
Desviación absoluta de la mediana normalizada (mad_normal)	868.772506
Coeficiente de variación (coef_var)	0.437485
Rango (range)	11760
Valor máximo (max)	12050
Valor mínimo (min)	290
Sesgo (skew)	1.370859
Curtosis (kurtosis)	7.166622
Test de normalidad de Jarque-Bera (jarque_bera)	17922.347382
Valor p del test de normalidad de Jarque-Bera (jarque_bera_pval)	0
Moda (mode)	1400
Frecuencia de la moda (mode_freq)	0.006304
Mediana (median)	1910
Percentil 0.1 %	522.890000
Percentil 1 %	720
Percentil 5 %	940
Percentil 25 %	1430
Percentil 75 %	2540
Percentil 95 %	3740
Percentil 99 %	4921.100000
Percentil 99.9 %	6965.550000

En la Tabla 3.7 Métricas para campos categóricos se muestran los datos calculados para campos categóricos.

Tabla 3.7: Métricas para campos categóricos

Nombre del campo (name)	waterfront
Valores del Top 5 (top5)	[0 1]
Frecuencia Top 5 (top5_freq)	[17166 124]
Probabilidades de Top 5 (top5_prob)	[0.99282822 0.00717178]
Elementos observados (nobs)	17290.0
Nulos (missing)	17290.0

En el Código 3 Mostrando Scores Promedios Calculados, se muestra cómo se calcula y se muestra el Score promedio para una selección específica de modelos. El código utiliza la función "sort_values"para ordenar los resultados en orden descendente según el puntaje. Luego, se filtran los resultados para incluir solo los modelos seleccionados y las columnas que muestran el puntaje y la Distancia al registro más cercano (DCR) en los tres umbrales *Synthetic vs Train* (ST), *Synthetic vs Hold* (SH) y *Train vs Hold* TH.

```
avg = syn.scores[syn.scores["type"] == "avg"]
avg.sort_values("score", ascending=False).loc[
    ["tddpm_mlp_21613","smote-enc_21613","gaussiancopula_noise_21613","tvae_21613",
    "tvae_noise_21613","gaussiancopula_21613",
    "copulagan_noise_21613","copulagan_21613","ctgan_noise_21613","ctgan_21613"],
    ["score", "DCR ST 5th", "DCR SH 5th", "DCR TH 5th"]]
```

Código 3: Mostrando Scores Promedios Calculados

El Score calculado se obtiene a través de SDV y se basa en cuatro métricas: KSComplement, TVComplement que conforman *Column Shapes*, ContingencySimilarity y CorrelationSimilarity conforman *Column Pair Trends*. Además, para mostrar los resultados, se proporciona un ejemplo de código en el Código 3 Mostrando Scores Promedios Calculados y un ejemplo de resultado en la Tabla 3.8 Ejemplo de scores promedios.

Tabla 3.8: Ejemplo de scores promedios

Nombre	Column Pair Trends	Column Shapes	**Score** ↓	DCR ST	DCR SH	DCR TH
ntddpm_mlp_21613	0.954	0.971	0.962	0.084	0.104	0.035
nsmote-enc_21613	0.941	0.967	0.954	0.058	0.090	0.035
<model>_<use_noise> _<n_sample>	0.941	0.967	0.954	0.058	0.090	0.035

Capítulo 4

Resultados

En este proyecto, se han generado datos sintéticos a través de diferentes preprocesamientos y modelos de machine learning. Los resultados obtenidos se presentan en términos de desempeño de los modelos entrenados con los datos sintéticos, así como en términos de similitud, privacidad y utilidad de los datos generados.

Es importante tener en cuenta que los resultados son específicos para cada conjunto de datos y modelo utilizado. Por lo tanto, se proporciona una descripción detallada de los resultados para cada caso. Esto permitirá una mejor comprensión de la eficacia de los diferentes métodos utilizados para generar datos sintéticos y cómo se comparan con los datos reales.

4.1. King County

4.1.1. Reportes

Scores

La Tabla 4.1 muestra los scores obtenidos para diferentes modelos utilizados en el proyecto. Como se puede apreciar, los modelos con puntajes más altos, como tddpm_mlp_21613 y smote-enc_21613, tienen un mayor parecido con el conjunto real. En cambio, los modelos con puntajes más bajos, como ctgan_noise_21613 y ctgan_21613, tienen una similitud muy baja con los datos reales.

Tabla 4.1: Scores King County

Nombre	Column Pair Trends	Column Shapes	**Score**
tddpm_mlp_21613	0.954061	0.971517	0.962789
smote-enc_21613	0.941977	0.967207	0.954592
gaussiancopula_noise_21613	0.845585	0.809726	0.827655
tvae_21613	0.781927	0.856533	0.819230
tvae_noise_21613	0.770583	0.859122	0.814853
gaussiancopula_21613	0.833007	0.792259	0.812633
copulagan_noise_21613	0.761719	0.844915	0.803317
copulagan_21613	0.745335	0.815264	0.780299
ctgan_noise_21613	0.749491	0.770946	0.760218
ctgan_21613	0.735844	0.746231	0.741038

Este resultado se puede apreciar en el Anexo B Lista completa de figura pairwise kingcounty, donde se muestran las diferencias entre los datos reales y los datos generados por cada modelo. Se puede observar que, en general, los modelos con puntajes más altos tienen una mayor similitud visual con los datos reales. Por ejemplo, las imágenes Figura 4.1 Comparación de conjunto Real y copulagan_noise_21613 (0.80) y Figura 4.2 Comparación de conjunto Real y gaussiancopula_21613 (0.81) muestran la comparación entre los datos reales y los datos generados por los modelos gaussiancopula_21613 y copulagan_noise_21613. A pesar de que estos modelos tienen puntajes similares, el modelo gaussiancopula_21613 tiene una mayor similitud visual con los datos reales que el modelo copulagan_noise_21613.

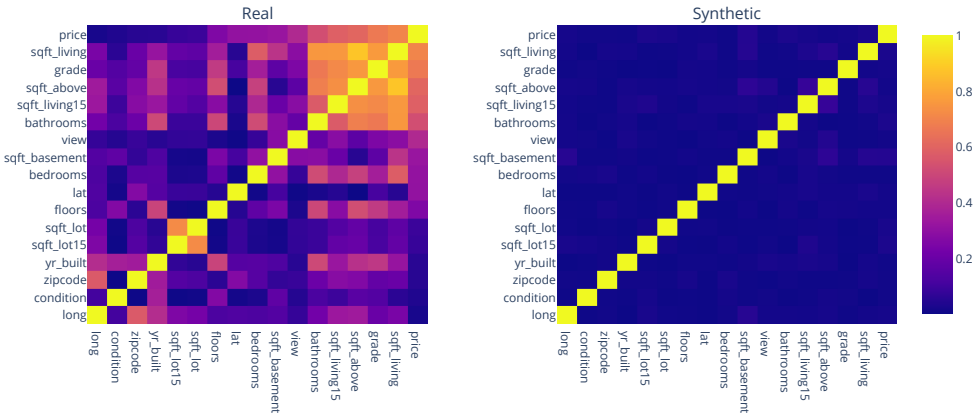

Figura 4.1: Comparación de conjunto Real y copulagan_noise_21613 (0.80)

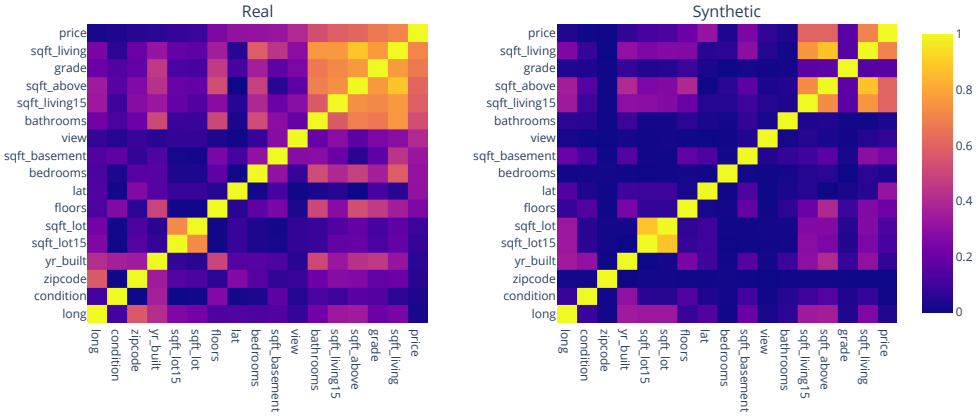

Figura 4.2: Comparación de conjunto Real y gaussiancopula_21613 (0.81)

Es importante destacar que, entre los modelos con puntajes superiores al 90 %, puede ser difícil evaluar visualmente cuál es el mejor. Esto se debe a que, a medida que el puntaje aumenta, la similitud visual entre los datos reales y los datos generados también aumenta. Esto se puede observar en las figuras Figura 4.3 Comparación de conjunto Real y smote-enc_21613 (0.95) y Figura 4.4 Comparación de conjunto Real y tddpm_mlp_21613 (0.96), donde se comparan los datos reales con los datos generados por los modelos smote-enc_21613 y tddpm_mlp_21613, respectivamente. Ambos modelos tienen puntajes superiores al 90 %, y la similitud visual entre los datos reales y los datos generados es muy alta en ambos casos.

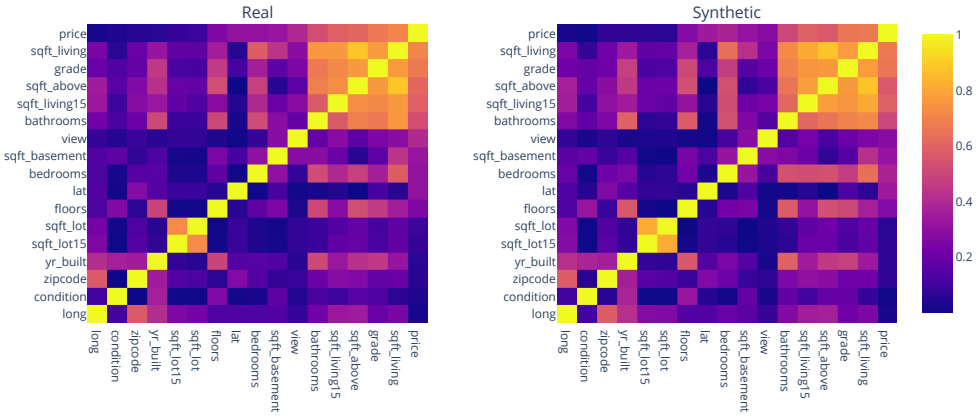

Figura 4.3: Comparación de conjunto Real y smote-enc_21613 (0.95)

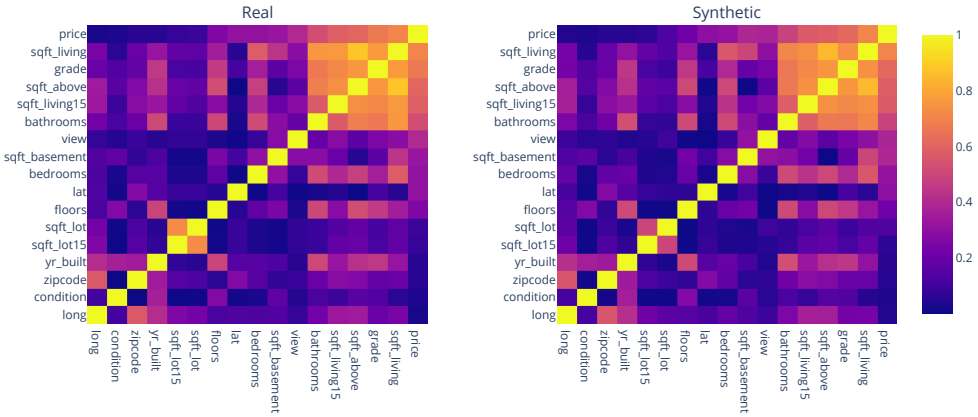

Figura 4.4: Comparación de conjunto Real y tddpm_mlp_21613 (0.96)

En la evaluación de SDMetrics y en la comparación visual utilizando la correlación de Wise, los mejores modelos encontrados son TDDPM y SMOTE. Estos modelos han obtenido los puntajes más altos en ambas métricas y también se han demostrado tener una mayor similitud visual con los datos reales. Por lo tanto, se puede concluir que estos modelos son los más efectivos para generar datos sintéticos útiles para este conjunto de datos específico.

Adhesión

Privacidad

Capítulo 5

Conclusiones

Capítulo 6

Discusión

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Definición 6.1 (ver [**?**]) *Definición definitiva*

$$\frac{\mathrm{d}}{\mathrm{d}x}\int_a^x f(y)\mathrm{d}y = f(x).$$

Bibliografía

[1] DALL·e 2.

[2] Imagen: Text-to-image diffusion models.

[3] Microsoft and google are in a 'game of thrones' battle over a.i.— but apple and amazon still have huge roles to play, according to wedbush.

[4] Papers with code - ImageNet benchmark (image classification).

[5] Stable diffusion public release.

[6] Angeela Acharya, Siddhartha Sikdar, Sanmay Das, and Huzefa Rangwala. GenSyn: A multi-stage framework for generating synthetic microdata using macro data sources.

[7] Accountability Act. Health insurance portability and accountability act of 1996. 104:191.

[8] Kiran Adnan and Rehan Akbar. An analytical study of information extraction from unstructured and multidimensional big data. 6:1–38. Publisher: Springer.

[9] Akim. TabDDPM: Modelling tabular data with diffusion models. original-date: 2022-10-02T23:01:07Z.

[10] Ewa Andrejczuk, Julian Martin Eisenschlos, Francesco Piccinno, Syrine Krichene, and Yasemin Altun. Table-to-text generation and pre-training with TabT5.

[11] Vadim Borisov, Kathrin Seßler, Tobias Leemann, Martin Pawelczyk, and Gjergji Kasneci. Language models are realistic tabular data generators.

[12] Peter Bruce, Andrew Bruce, and Peter Gedeck. *Practical statistics for data scientists: 50+ essential concepts using R and Python*. O'Reilly Media.

[13] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. SMOTE: synthetic minority over-sampling technique. 16:321–357.

[14] Sabrina De Capitani Di Vimercati, Sara Foresti, Giovanni Livraga, and Pierangela Samarati. Data privacy: Definitions and techniques. 20(6):793–817. Publisher: World Scientific.

[15] Khaled El Emam, Lucy Mosquera, and Richard Hoptroff. *Practical synthetic data generation: balancing privacy and the broad availability of data*. O'Reilly Media.

[16] John Gantz and David Reinsel. The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. 2007(2012):1–16.

[17] Jonathan Herzig, Pawe{\textbackslash}l Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Martin Eisenschlos. TaPas: Weakly supervised table parsing via pre-training.

[18] HARLFOXEM Kaggle. House sales in king county, USA.

[19] Mihir Kale and Abhinav Rastogi. Text-to-text pre-training for data-to-text tasks.

[20] Akim Kotelnikov, Dmitry Baranchuk, Ivan Rubachev, and Artem Babenko. Overview — SDV 0.18.0 documentation.

[21] Akim Kotelnikov, Dmitry Baranchuk, Ivan Rubachev, and Artem Babenko. TabDDPM: Modelling tabular data with diffusion models.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc.

[23] Dan Milmo and Dan Milmo Global technology editor. Google v microsoft: who will win the AI chatbot race?

[24] OpenAI. ChatGPT: a large language model trained by OpenAI.

[25] Stuart L Pardau. The california consumer privacy act: Towards a european-style privacy regime in the united states. 23:68. Publisher: HeinOnline.

[26] Neha Patki, Roy Wedge, and Kalyan Veeramachaneni. The synthetic data vault. In *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 399–410. IEEE.

[27] David Pujol, Amir Gilad, and Ashwin Machanavajjhala. PreFair: Privately generating justifiably fair synthetic data.

[28] Protection Regulation. Regulation (EU) 2016/679 of the european parliament and of the council. 679:2016.

[29] Aivin V Solatorio and Olivier Dupriez. REaLTabFormer: Generating realistic relational and tabular data using transformers.

[30] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Modeling tabular data using conditional gan. 32.

[31] Zilong Zhao, Aditya Kunar, Robert Birke, and Lydia Y. Chen. CTAB-GAN: Effective table data synthesizing. In Vineeth N. Balasubramanian and Ivor Tsang, editors, *Proceedings of The 13th Asian Conference on Machine Learning*, volume 157 of *Proceedings of Machine Learning Research*, pages 97–112. PMLR.

[32] Zilong Zhao, Aditya Kunar, Robert Birke, and Lydia Y Chen. CTAB-GAN+: Enhancing tabular data synthesis.

Apéndice A

Código de entrenamiento de economicos

```python
import pandas as pd
from syntheticml.data.synthetic import Synthetic, MODELS
from syntheticml.models.tab_ddpm.sdv import SDV_MLP
import torch
import numpy as np
import itertools
import multiprocessing as mp
import os

def test_train(args):
    lrc, ntc, sts, btsc, rtdlc, syn, df = args
    #notebooks/economicos_good/2e-06_10_100000_5000_1024-512-256
    checkpoint = "economicos_good2/" +  "_".join(
            map(str, [lrc, ntc, sts, btsc, "-".join(map(str, rtdlc))]))
    checkpoint = "con_fechas"
    if os.path.exists(f"{checkpoint}/final_model.pt") or os.path.exists(f"{checkpoint}/exit"):
        return (checkpoint, 1)
    model = SDV_MLP(syn.metadata,
                    "_price",
                    exclude_columns=syn.exclude_columns,
                    df=df,
                    batch_size=btsc,
                    steps=sts,
                    checkpoint=checkpoint,
                    num_timesteps=ntc,
                    weight_decay=0.0,
                    lr=lrc,
                    model_params=dict(rtdl_params=dict(
                        dropout=0.0,
                        d_layers=rtdlc
                    ))
                    )
    model.fit(syn.train)
    model.save(f"{checkpoint}/final_model.pt")
    return (checkpoint, 1)

if __name__ == '__main__':
    df = pd.read_parquet('../datasets/economicos/synth/split/train.parquet')
    category_columns=("property_type", "transaction_type", "state", "county", "rooms", "bathrooms", "m_built", "m_size", "source", )
    # TODO: Estudiar implicancia de valores nulos en categorias y numeros
    df_converted = df.astype({k: 'str' for k in ("description", "price", "title", "address", "owner",)})
    basedate = pd.Timestamp('2017-12-01')
    dtime = df_converted.pop("publication_date")
    df_converted["publication_date"] = dtime.apply(lambda x: (x - basedate).days)
    syn = Synthetic(df_converted,
            id="url",
            category_columns=category_columns,
            text_columns=("description", "price", "title", "address", "owner",),
            exclude_columns=tuple(),
            synthetic_folder = "../datasets/economicos/synth",
            models=['copulagan', 'tvae', 'gaussiancopula', 'ctgan', 'smote-enc'],
            n_sample = df.shape[0],
            target_column="_price"
    )

    lrs = np.linspace(2e-6, 2e-3, 10)
    num_timesteps = np.linspace(10, 1000, 3, dtype=int)
    batch_size = np.linspace(2500, 5000, 3, dtype=int)
    steps = np.linspace(150000, 500000, 5, dtype=int)
    rtdl_params = [
        [1024, 512, 256], [512, 256], [256, 128], [256, 128, 128], [256, 128, 128, 128]
```

Apéndice B

Lista completa de figura pairwise kingcounty

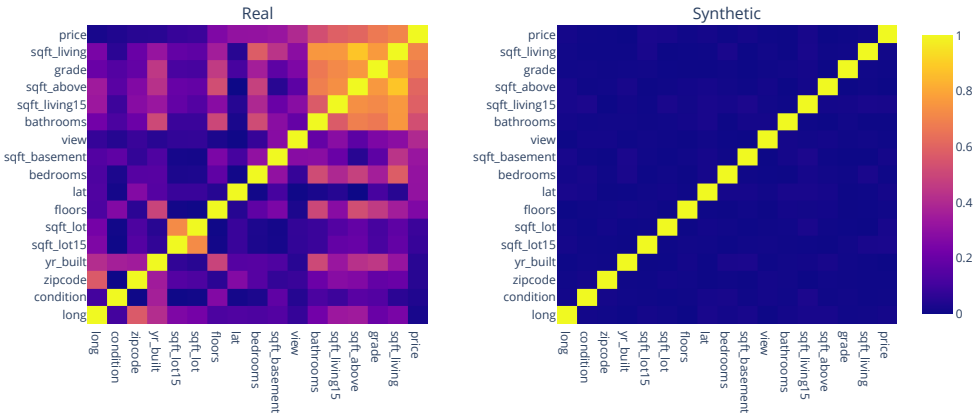

Figura B.1: Comparación de conjunto Real y Modelo: ctgan_21613

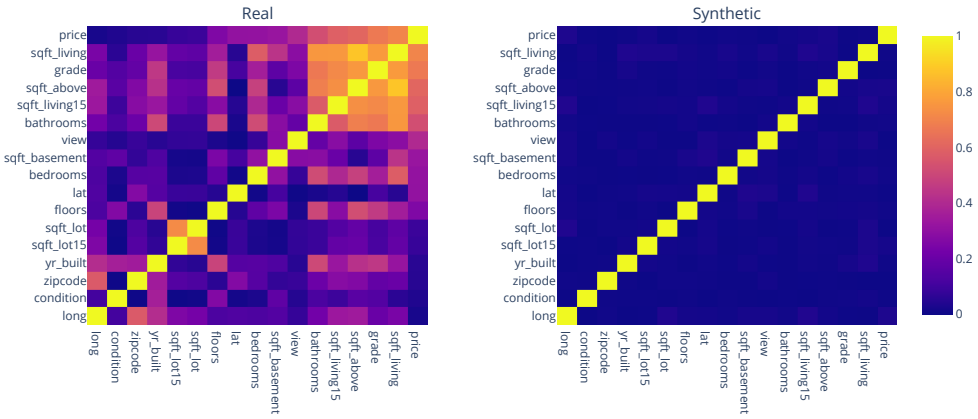

Figura B.2: Comparación de conjunto Real y Modelo: ctgan_noise_21613

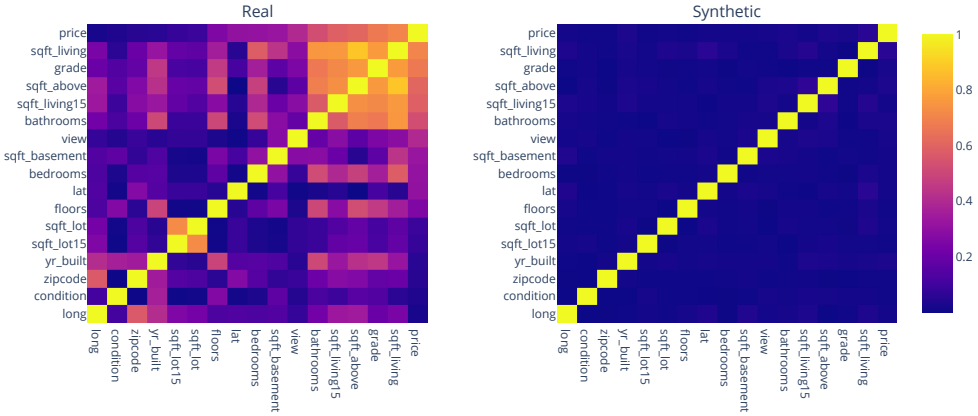

Figura B.3: Comparación de conjunto Real y Modelo: copulagan_21613

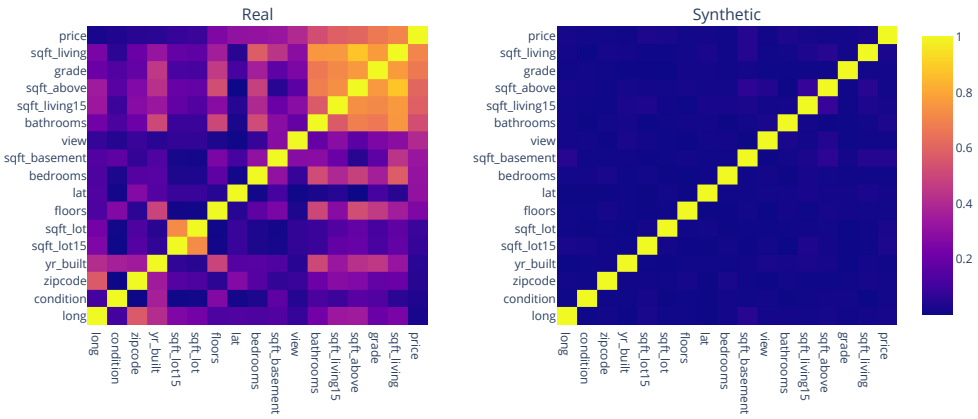

Figura B.4: Comparación de conjunto Real y Modelo: copulagan_noise_21613

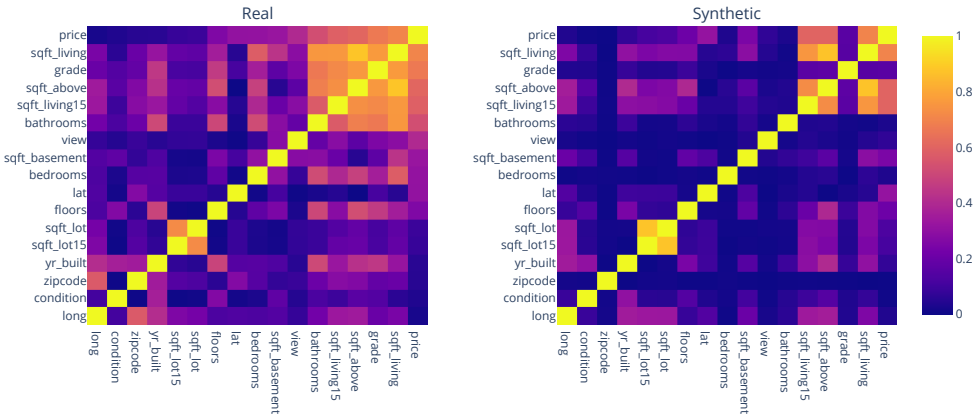

Figura B.5: Comparación de conjunto Real y Modelo: gaussiancopula_21613

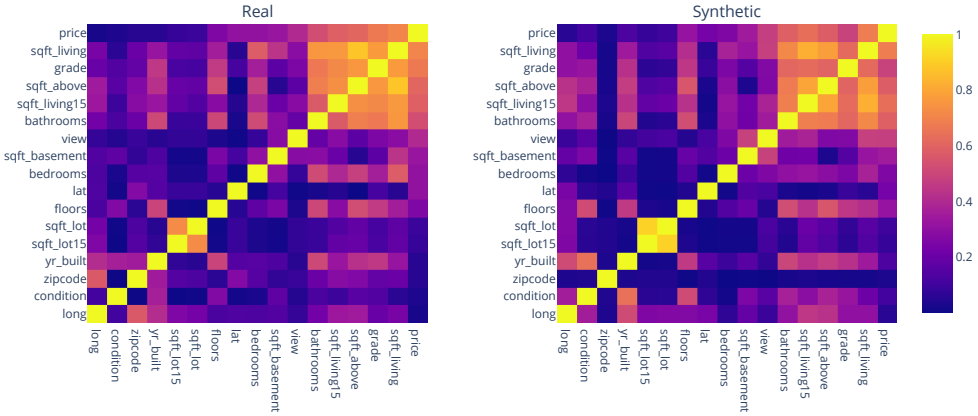

Figura B.6: Comparación de conjunto Real y Modelo: tvae_noise_21613

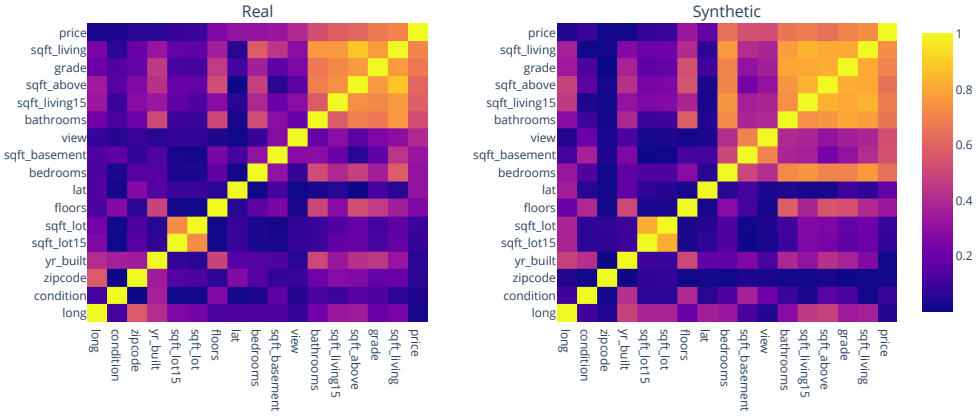

Figura B.7: Comparación de conjunto Real y Modelo: tvae_21613

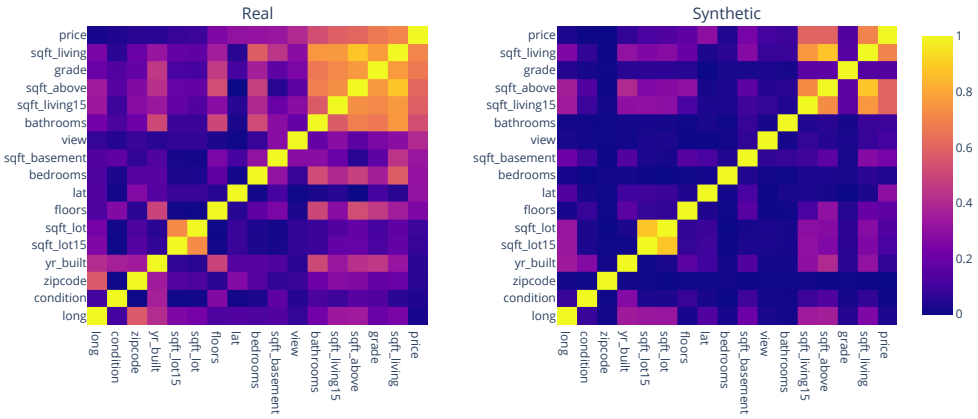

Figura B.8: Comparación de conjunto Real y Modelo: gaussiancopula_noise_21613

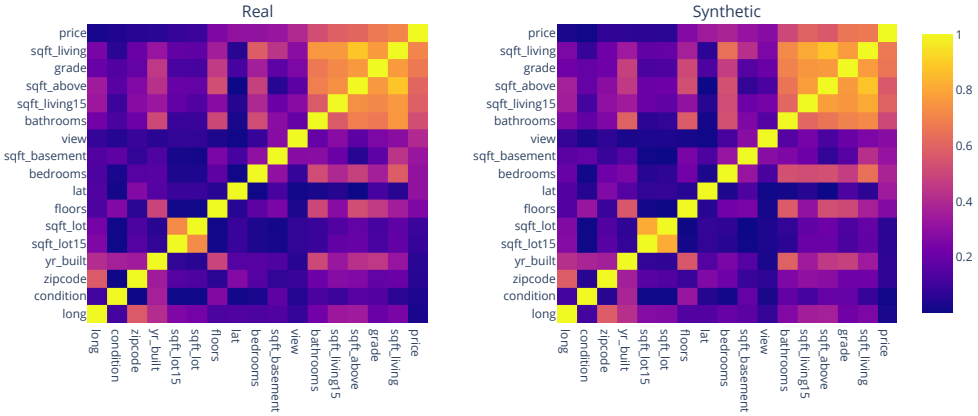

Figura B.9: Comparación de conjunto Real y Modelo: smote-enc_21613

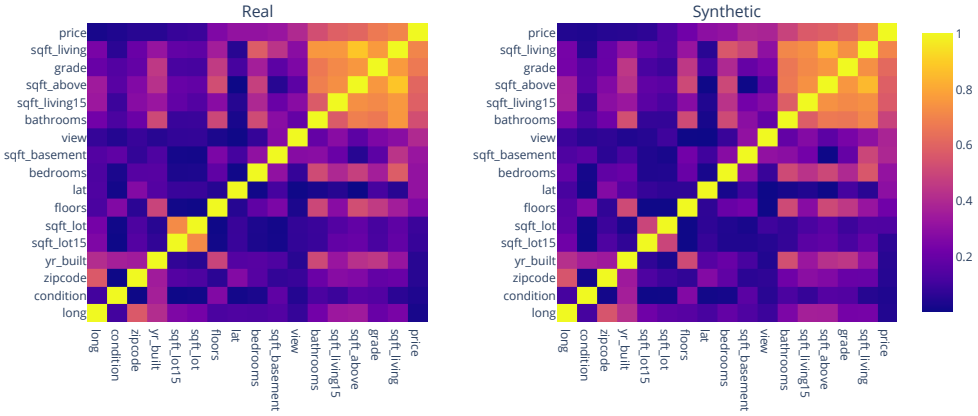

Figura B.10: Comparación de conjunto Real y Modelo: tddpm_mlp_21613