TECNOLÓGICO SUPERIOR DE PURÍSIMA DEL RINCÓN

Nombres del Alumno (a):

Alva Zavala Gabriela – RS21110105

Botello Santos Rufino – RS21110220

Márquez Robles Emmanuel – RS21110007

Medina Martínez Oscar Eduardo - RS21110201

Carrera:

Ing. Informática

Grado y grupo:

7-A

Materia:

Inteligencia de Negocios

Nombre del docente:

Valentín Calzada Ledesma

Data Mining con Algoritmos ML.

Purísima del Rincón, Gto. A 28 de octubre de 2024

1. Introducción

En el presente documento se muestra una evaluación sobre siete algoritmos de Machine Learning de aprendizaje supervisado. Estos siete algoritmos son:

- **Support Vector Machines (SVM).** Es un clasificador que busca el hiperplano óptimo que separa diferentes clases en un espacio multidimensional.
- **Stochastic Gradient Descent (SGD).** Es un algoritmo de optimización que actualiza los parámetros de un modelo en función de un subconjunto aleatorio de datos (mini-batch) en lugar de usar todo el conjunto de datos a la vez.
- **Nearest Neighbors.** Es un algoritmo de clasificación que asigna una clase a un nuevo punto basándose en la clase de sus k vecinos más cercanos en el espacio de características.
- **Gaussian Process.** Es un método no paramétrico que se utiliza para modelar distribuciones sobre funciones. Se basa en la suposición de que los datos pueden ser descritos por una función continua.
- **Naive Bayes.** Este es un conjunto de algoritmos de clasificación basados en el teorema de Bayes y la suposición de independencias entre las características.
- **Decision Trees.** Son estructuras en forma de árbol que dividen el conjunto de datos en subconjuntos basados en características. Cada nodo representa una prueba en una característica, cada rama representa el resultado de la prueba, y cada hoja representa una clase.
- **Neural Networks (Multi.Layer Perceptron (MLP)).** Son un tipo de red neuronal compuesta por múltiples capas de nodos (neuronas) que pueden aprender representaciones complejas a partir de los datos.

Para evaluar la precisión con la que pueden trabajar estos algoritmos se seleccionaron tres datasets diferentes del repositorio de UC Irvine Machine Learning Repository (UCI) y utilizando el Cross Validation. Estos datasets son:

- **Predict Students Dropout and Academic Success.** Este dataset contiene variables relacionadas con el rendimiento académico, asistencia, factores socioeconómicos y participación estudiantil; lo cual ayuda a predecir la probabilidad de abandono escolar y medir el éxito académico de los estudiantes.
- **Early Stage Diabetes Risk Prediction.** Este dataset contiene variables relacionadas con los síntomas de la diabetes como la sed, fatiga, visión borrosa, pérdida de peso y niveles de azúcar; esto ayuda a la predicción de riesgo de diabetes en etapas tempranas con los síntomas para indicar el desarrollo de la enfermedad.
- **Maternal Health Risk.** Este dataset contiene variables relacionadas con la presión arterial, nivel de glucosa y estrés; en si se enfoca en evaluar el riesgo de salud en mujeres embarazadas. Considerando los síntomas las clasifica en riesgo bajo, medio y alto.
- **Cross Validation.** Es una técnica utilizada en el entrenamiento y evaluación de modelos de aprendizaje automático para medir su rendimiento y generalización a nuevos datos. Su principal objetivo es evitar el sobreajuste, que ocurre cuando un modelo se ajusta demasiado a los datos de entrenamiento, y por ende no se desempeña bien en datos no vistos.

2. Metodología

Importación de librerías.

Para poder realizar la evaluación de los algoritmos primero necesitamos descargar las librerías de Python de “scikit-learn”, pandas y si se necesita graficar matplotlib.

Importación de dataset.

Utilizamos pandas para cargar los dataset en un dataframe y que le sea más fácil

a Python entender el contenido. Al momento de cargar el dataset le indicamos con que signo de puntuación están separados los datos en el CSV.

Preprocesamiento de datos.

Para que el algoritmo no crashee primero debemos limpiar todos los datos. Por ejemplo, rellenar campos vacíos o que tengan Null en sus valores, esto implica rellenar campos numéricos o campos con tipos de datos que no pueda comprender el algoritmo. Para evitar crasheos en datos no numéricos se usan librerías de conversión a datos numéricos.

Aplicación de algoritmo.

Después de aplicar el preprocesamiento ya podemos implementar el algoritmo, simplemente le damos los datos que requiere el modelo, que por lo general son X e Y. Si se requiere, se pueden graficar las clasificaciones realizadas por el algoritmo o incluso la precisión con la que este trabajó.

Evaluación de precisión con Cross Validation.

Por último, aplicamos la librería de Cross Validation con los datos X e Y, el número de capas o pliegues y extraemos la precisión del modelo. Después imprimimos el Accuracy obtenido en base al Cross para identificar que tan bien trabaja el modelo.

3. Tabla comparativa y resultados

	Data Set		
Algoritmo	**Predict Students Dropout and Academic Success**	**Early Stage Diabetes Risk Prediction**	**Maternal Health Risk**
Support Vector Machines	0.7619 = 76.19%	0.9673 = 96.73%	0.6971 = 69.71%
Stochastic Gradient Descent	0.7425 = 74.25%	0.8846 = 88.46%	0.5569 = 55.69%
Nearest Neighbors	0.66 = 66%	0.92 = 92%	0.73 = 73%
Gaussian Process	0.68 = 68%	0.88 = 88%	0.45 = 45%
Naive Bayes	0.6779 = 67.79%	0.8769 = 87.69%	0.6026 = 60.26%
Decision Trees	0.6793 = 67.93%	0.9577 = 95.77%	0.8068 = 80.68%
Neural Networks (Multilayer Perceptron)	0.7028 = 70.28%	0.9712 = 97.12%	0.7031 = 70.31%

Predict Students Dropout and Academic Success

Support Vector Machines.

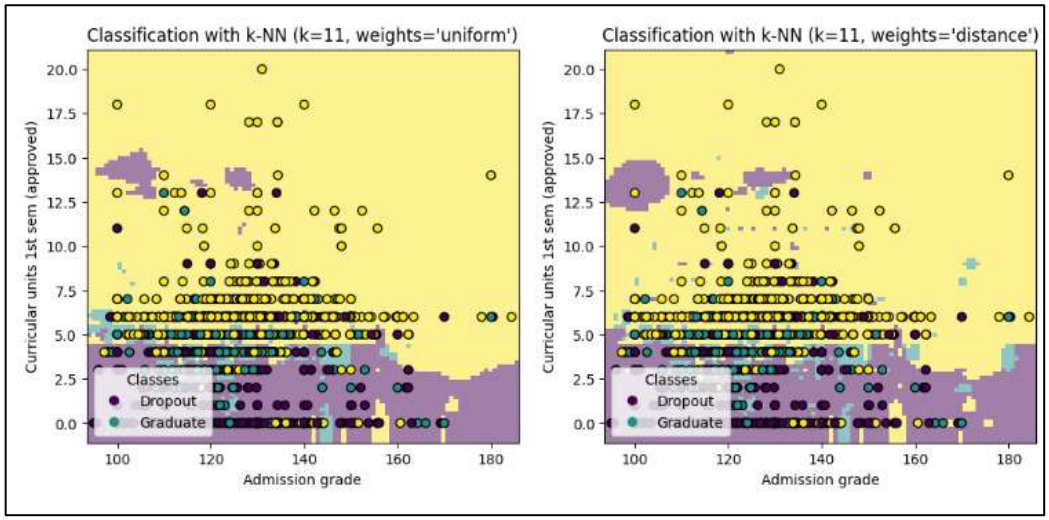
```
SVM precisión: 0.7619853324208655 Desviación estándar: 0.022215466847768595
```

Stochastic Gradient Descent.

```
SGD precisión: 0.742541597295282 Desviación estándar 0.015028756326286828
```

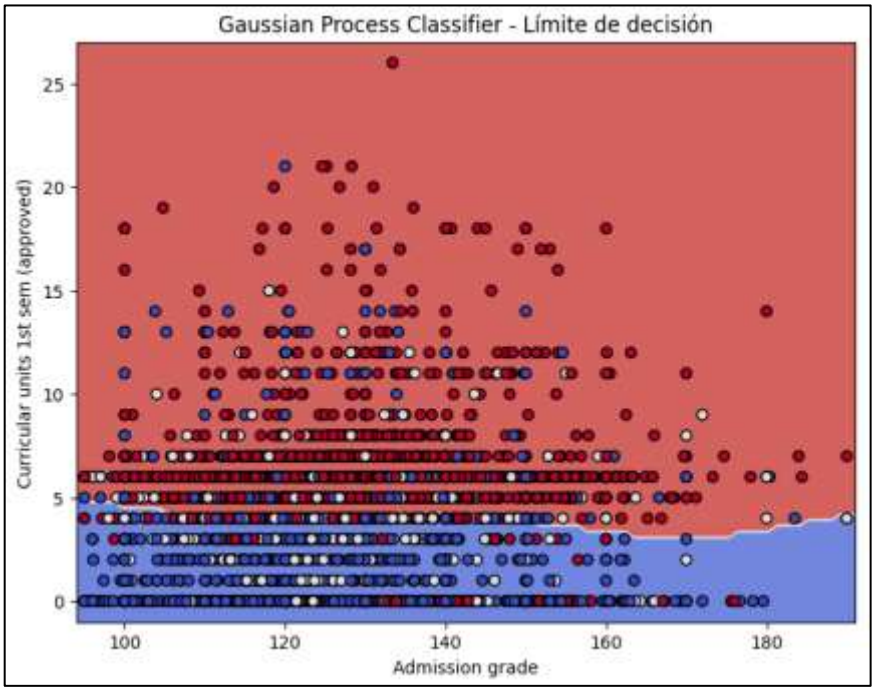
Nearest Neighbors.

```
Accuracy: 0.66
```



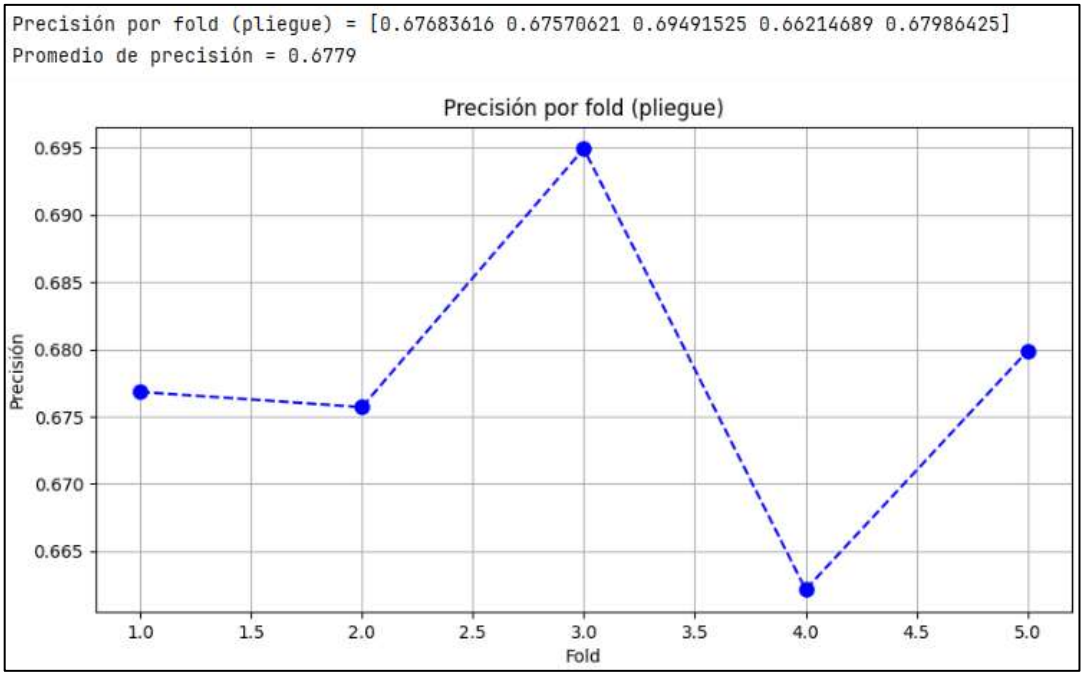
Gaussian Process.

```
Accuracy: 0.68
```



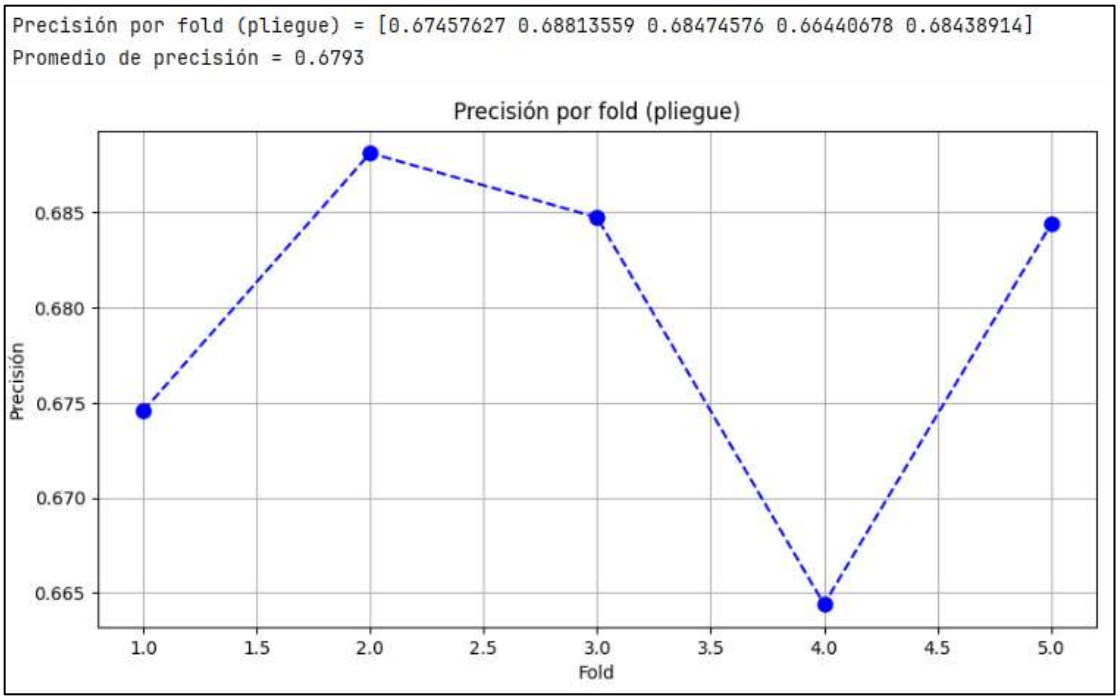
Naive Bayes.

```
Precisión por fold (pliegue) = [0.67683616 0.67570621 0.69491525 0.66214689 0.67986425]
Promedio de precisión = 0.6779
```

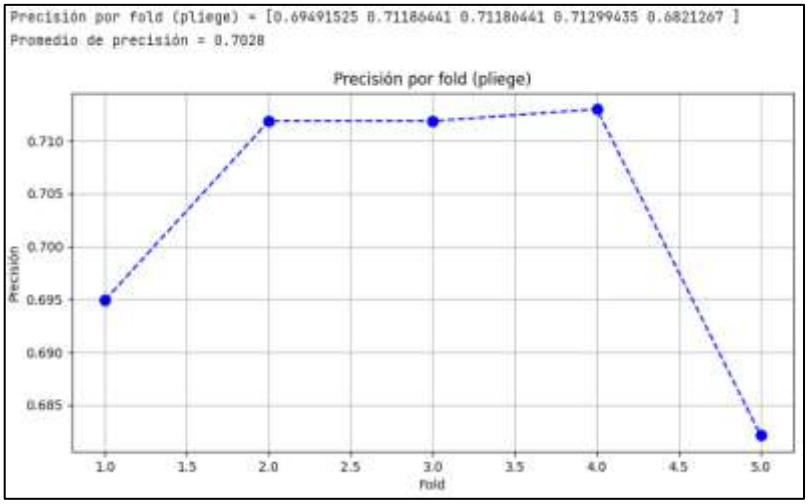


Decision Trees.

```
Precisión por fold (pliegue) = [0.67457627 0.68813559 0.68474576 0.66440678 0.68438914]
Promedio de precisión = 0.6793
```



Neural Networks (Multilayer Perceptron).



Early Stage Diabetes Risk Prediction

Support Vector Machines.

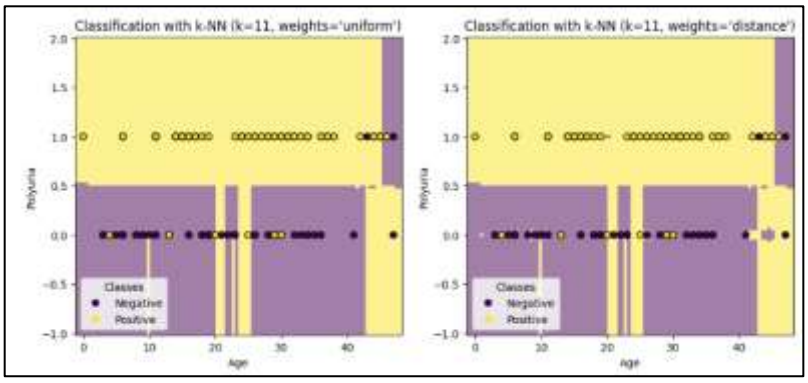
```
SVM precisión: 0.9673076923076922 Desviación estándar: 0.037536964030659876
```

Stochastic Gradient Descent.

```
SGD precisión 0.8846153846153845 Desviación estándar: 0.04865042554105201
```

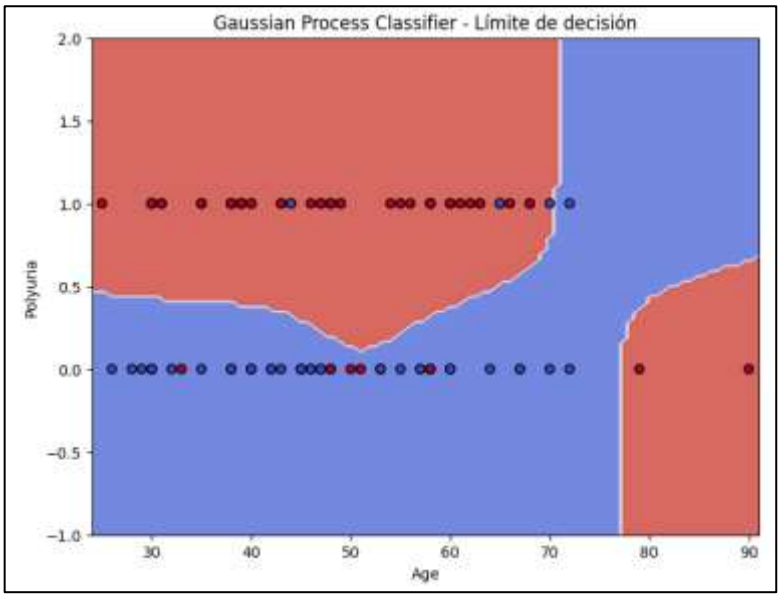
Nearest Neighbors.

```
Accuracy: 0.92
```



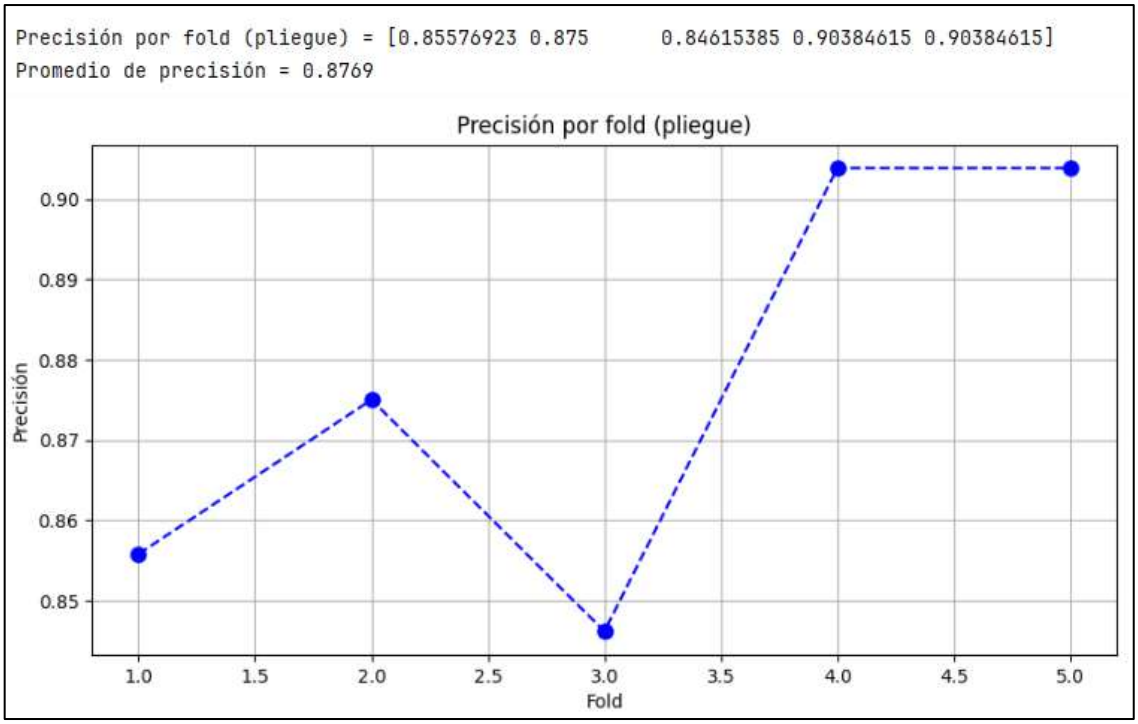
Gaussian Process.

```
Accuracy: 0.88
```



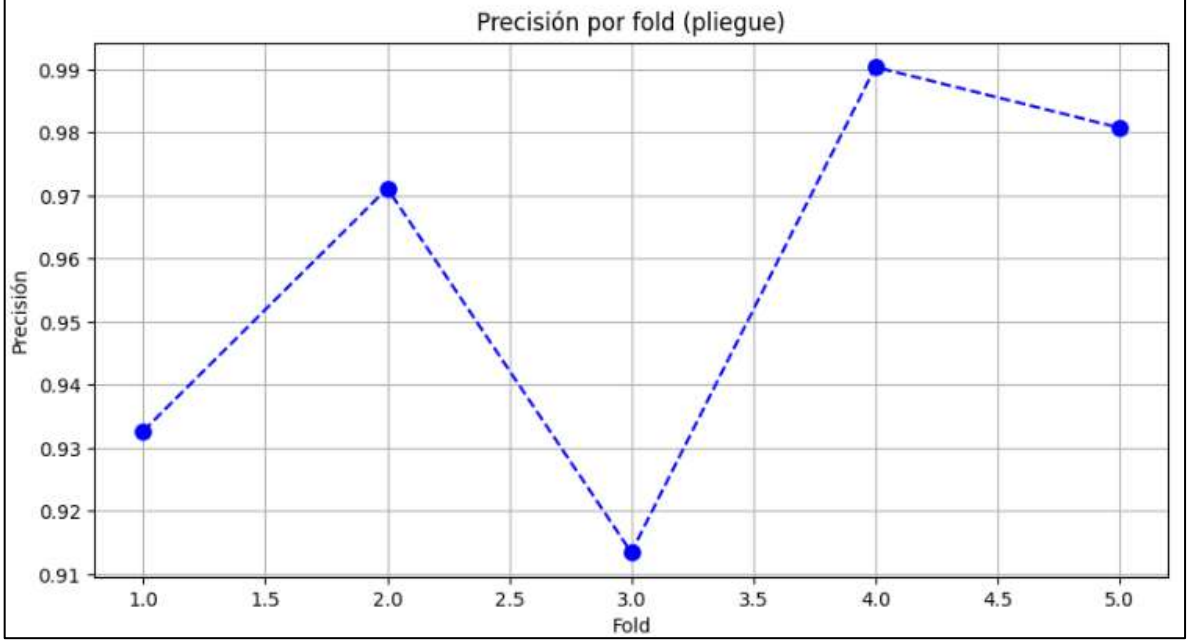
Naive Bayes.

```
Precisión por fold (pliegue) = [0.85576923 0.875      0.84615385 0.90384615 0.90384615]
Promedio de precisión = 0.8769
```



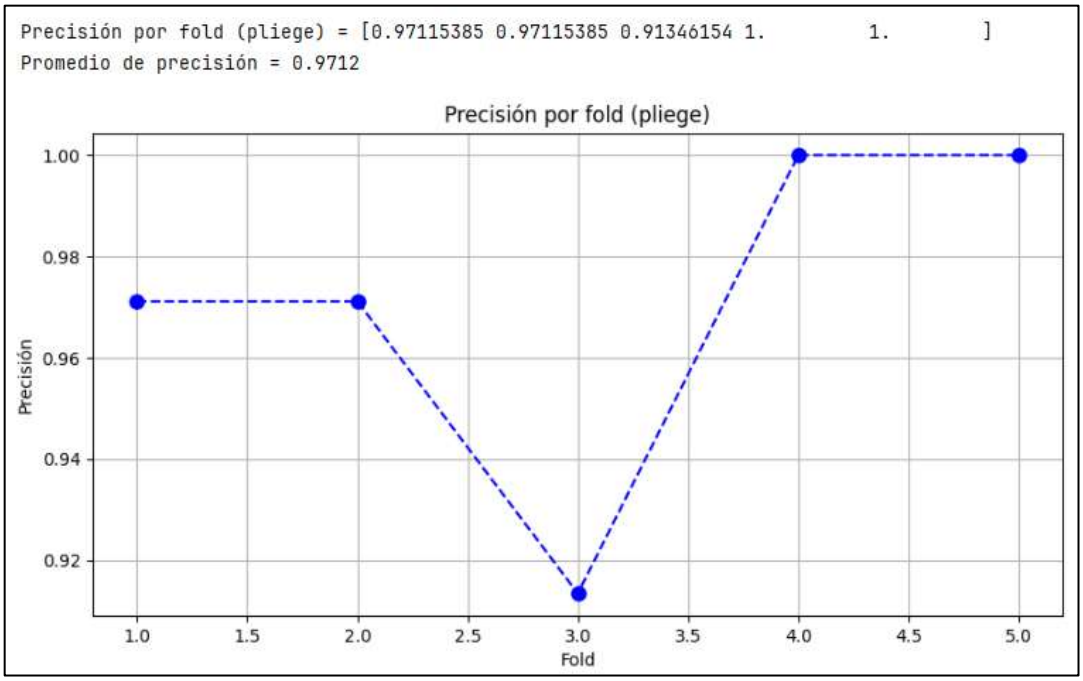
Decision Trees.

```
Precisión por fold (pliegue) = [0.93269231 0.97115385 0.91346154 0.99038462 0.98076923]
Promedio de precisión = 0.9577
```



Neural Networks (Multilayer Perceptron).

```
Precisión por fold (pliege) = [0.97115385 0.97115385 0.91346154 1.         1.        ]
Promedio de precisión = 0.9712
```



7A ING. INF
Materia: Inteligencia de Negocios
Data Mining con algoritmos ML

Maternal Health Risk

Support Vector Machines.

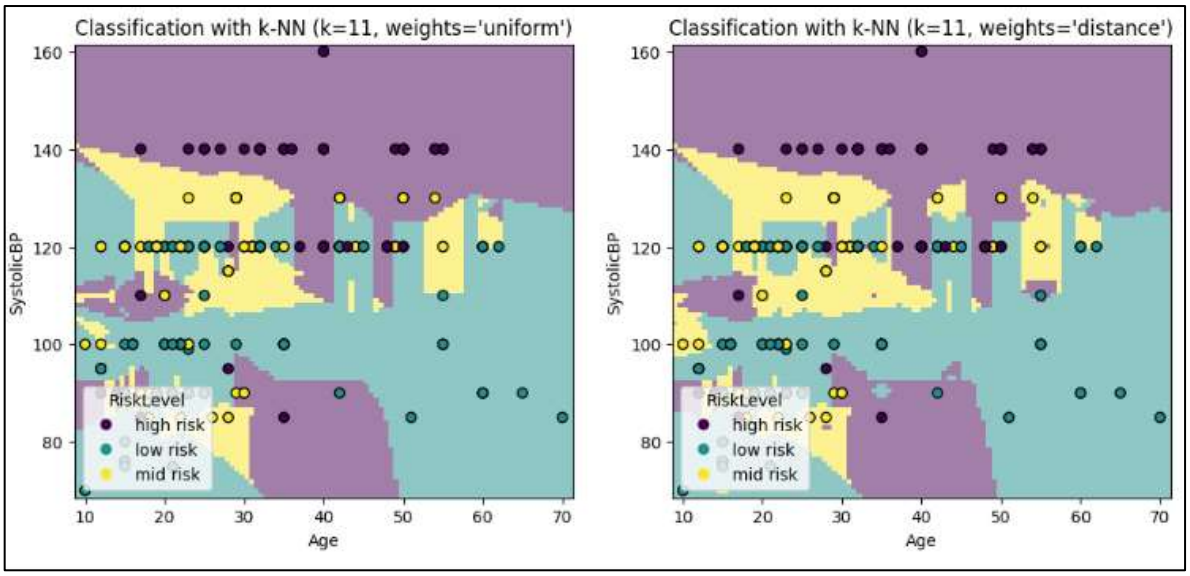
```
SVM precisión: 0.6971170646476412 Desviación estándar: 0.0695281095002784
```

Stochastic Gradient Descent.

```
SGD precisión 0.5569986410405747 Desviación estándar: 0.09238951718562731
```

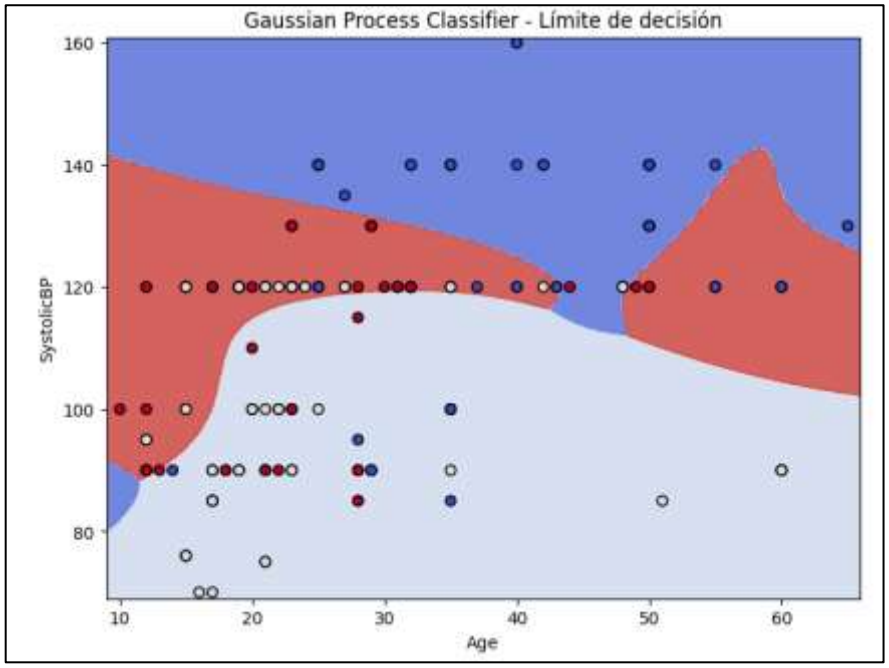
Nearest Neighbors.

```
Accuracy: 0.73
```



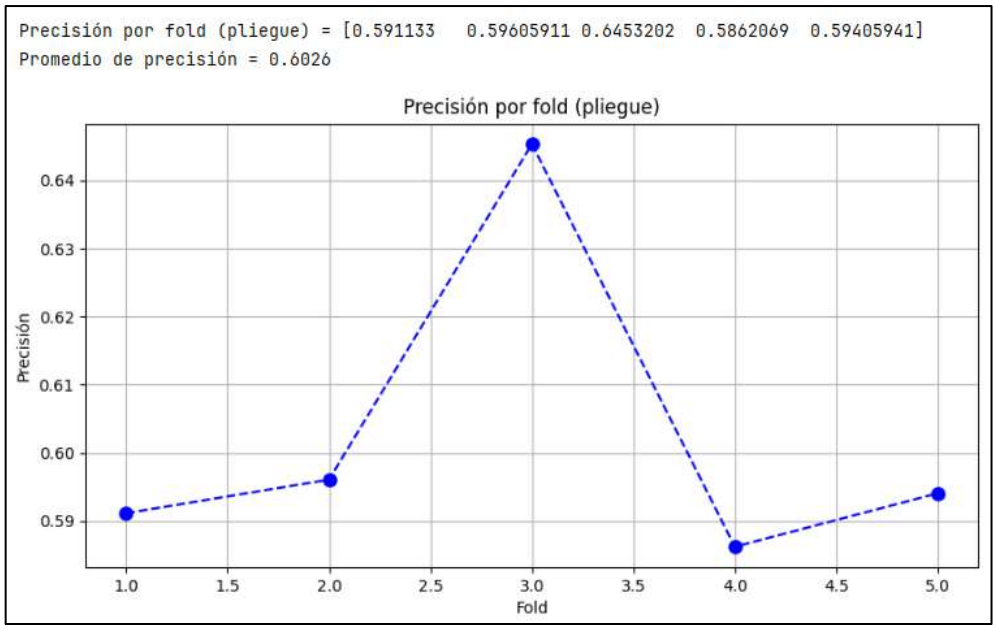
Gaussian Process.

```
Accuracy: 0.45
```



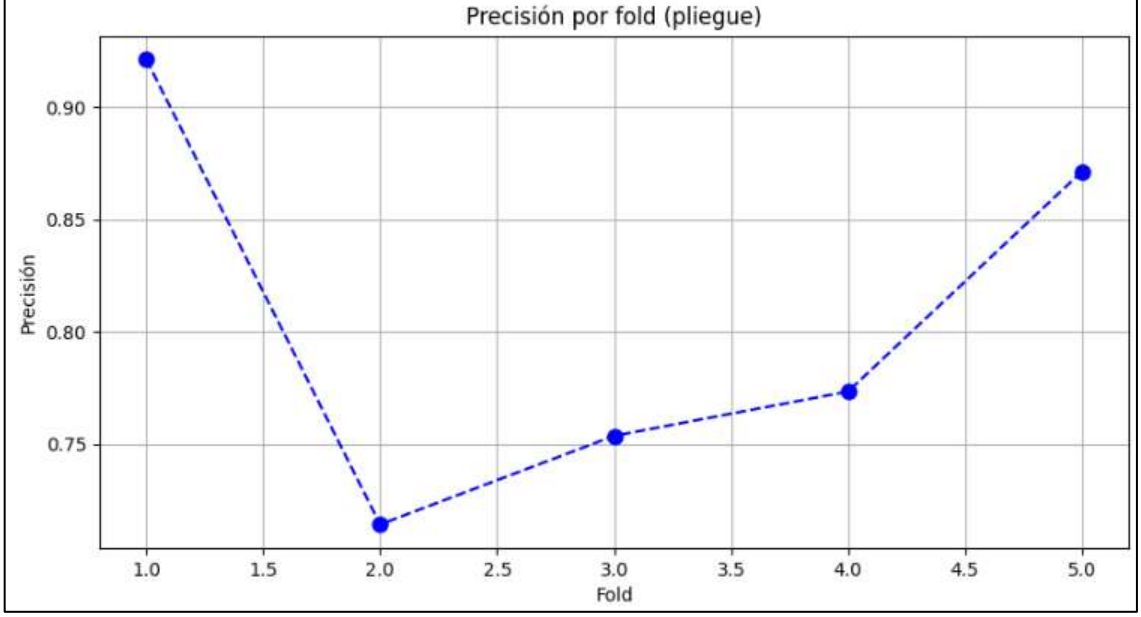
Naive Bayes.

```
Precisión por fold (pliegue) = [0.591133   0.59605911 0.6453202  0.5862069  0.59405941]
Promedio de precisión = 0.6026
```



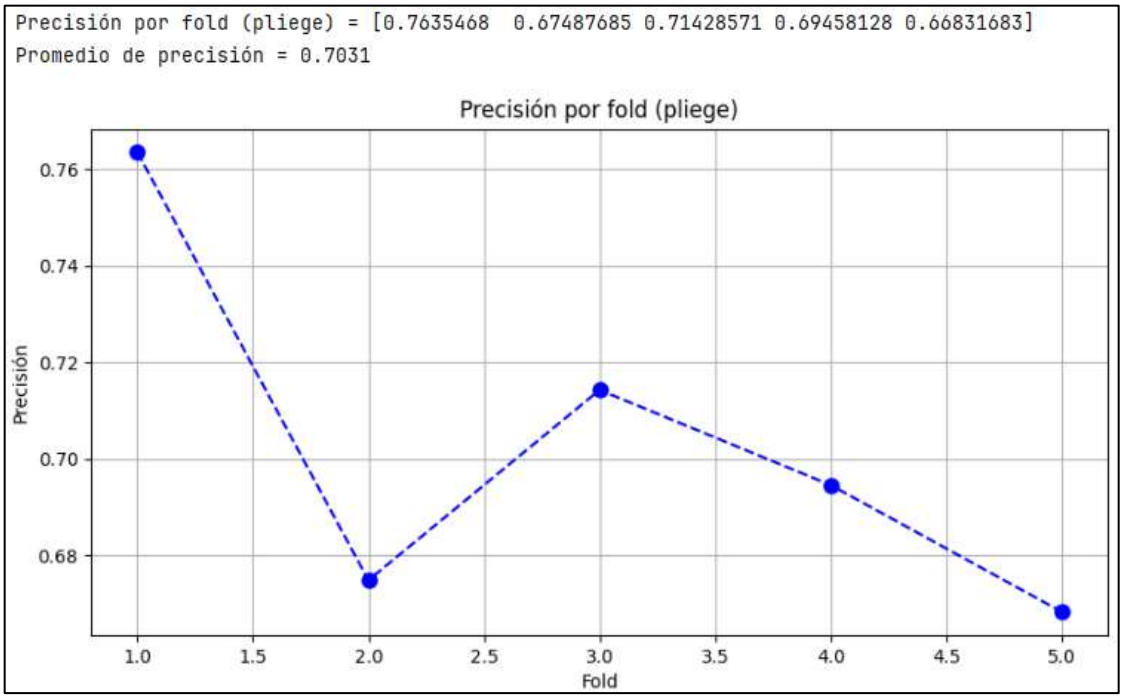
Decision Trees.

```
Precisión por fold (pliegue) = [0.92118227 0.71428571 0.75369458 0.77339901 0.87128713]
Promedio de precisión = 0.8068
```



Neural Networks (Multilayer Perceptron).

```
Precisión por fold (pliege) = [0.7635468  0.67487685 0.71428571 0.69458128 0.66831683]
Promedio de precisión = 0.7031
```



4. Conclusión

Tras haber realizado una prueba con cada algoritmo de Machine Learning sobre los tres data sets podemos concluir que el mejor algoritmo para el data set de Predict Students Dropout and Academic Success fue el Support Vector Machines con un 76% de precisión; para el data set de Early Stage Diabetes Risk Prediction fue el Neural Networks MPL con un 97% de precisión; y para el data set de Maternal Health Risk Data Set fue el Decision Trees con el 80% de precisión.

Los resultados pueden deberse a principales factores, por ejemplo los datos con los que trabaja cada algoritmo, los métodos de cada algoritmo o incluso tanto la aplicación del algoritmo como la aplicación del data set.

El Support Vectir Machines es excelente para la clasificación de datos complejos o con clases no lineales. La Neural Networks MLP son excelentes para la clasificación, regresión y reconocimiento de patrones. Los Decision Trees son excelentes para la clasificación y la interpretación, además de que por lo general de utilizan en la evaluación de riesgos y en la predicción en el campo de la medicina.