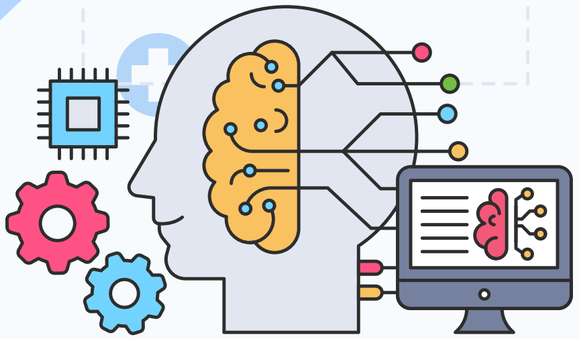
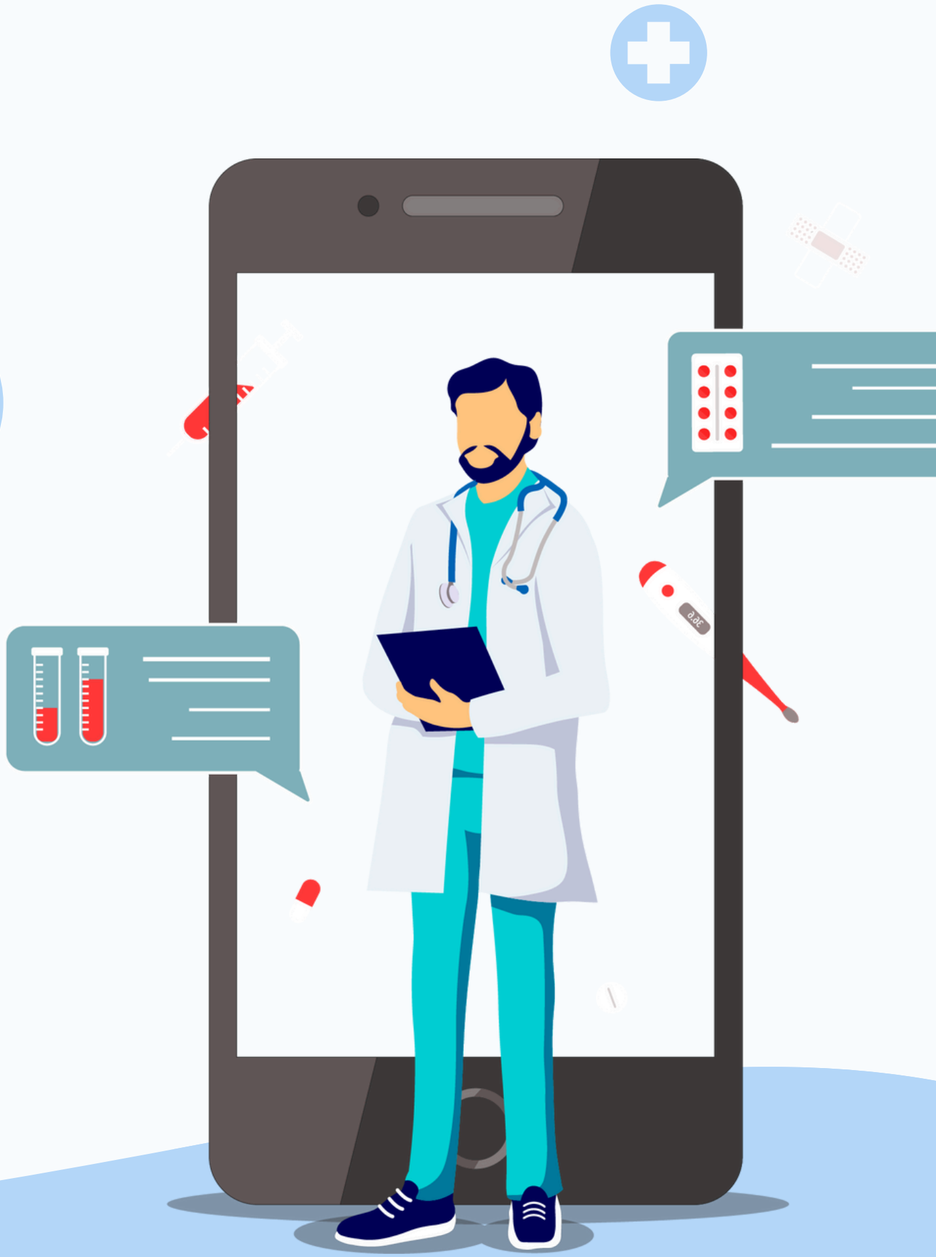
UNIVERSIDAD PERUANA
CAYETANO HEREDIA

Proyecto

- Álvaro Sevilla
- Diego Salvatierra
- Manuel Hernández



Contenido



Introducción

Proteger y promover la salud de los trabajadores en su entorno laboral

- Prevenir enfermedades
- Identificar riesgos
- Promover entornos seguros

Ley N° 30222

- Realizar exámenes médicos cada 2 años
- Trabajos de alto riesgo: antes, durante y al término.
- Gastos cubre el empleador

A nivel mundial:

2 mill muertes

160 mill enfermos

270 mill lesiones

Respiratorio y cardiovascular

4% PIB mundial

Perú:

274 mil nuevos puestos de trabajo cada año

[1]Jeny Flores, “EFECTIVIDAD DE UN PROGRAMA DE SALUD OCUPACIONAL FRENTE A LAS ACTITUDES DE TRABAJADORES EN LOS EXÁMENES MÉDICOS OCUPACIONALES DE UNA CLINICA PRIVADA.”, Tesis de Licenciatura, Univ. Peru. Cayetano Heredia, Lima, 2017.
[2] M. Angela y V. Linda, “Patologías asociadas a la actividad laboral: Una visión desde la salud ocupacional”, Dominio Cienc., 2022. [En línea]. Disponible: https://doi.org/10.23857/dc.v8i3

Factores

Horarios rotativos, carga laboral, posturas y contaminantes

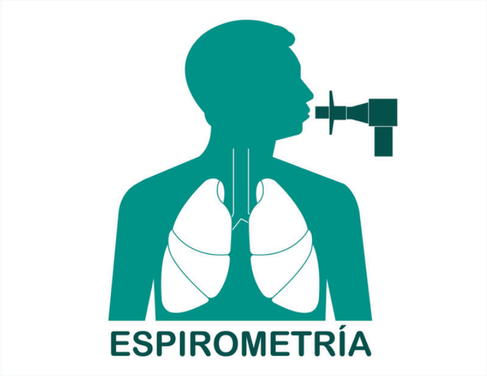
Efectos

Daño físico y psicológico, afecciones al sistema cardiovascular, respiratorio, etc.

Exámenes

Test psicológicos

Espirometría



Audiometría

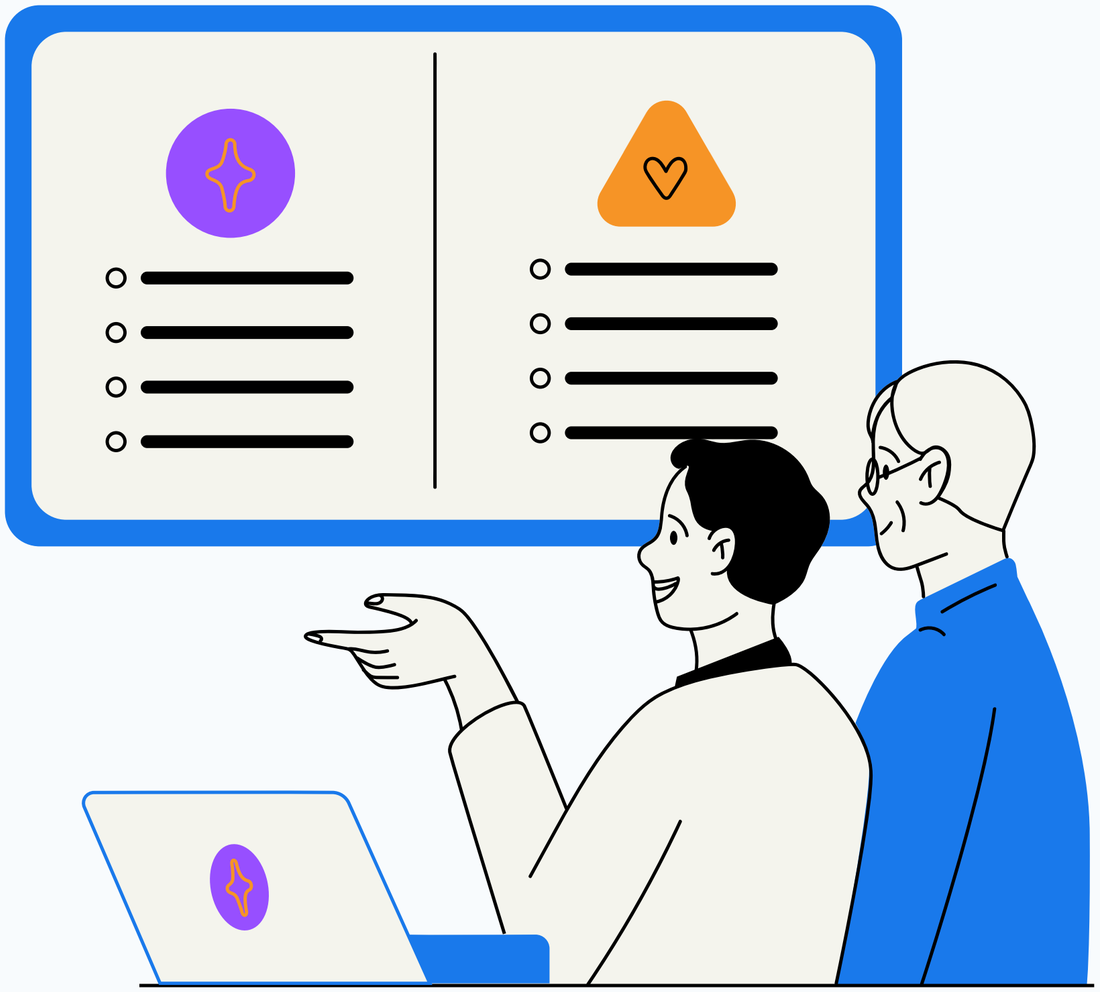
GRAN CANTIDAD DE DATOS

Problemática

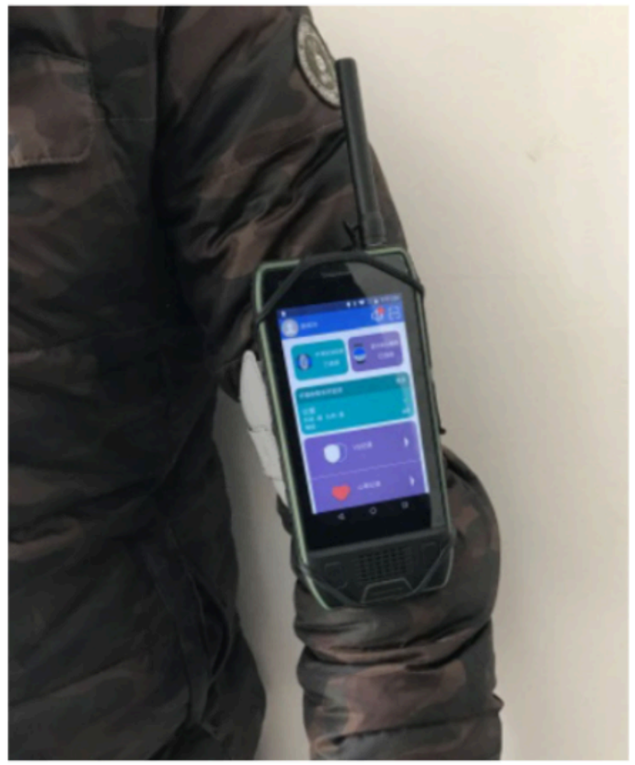
“Ausencia de herramientas que permitan automatizar la clasificacion de trabajadores basados en sus resultados de examenes de salud ocupacional”

- Inconsistencia de criterios de clasificación
- Dificultad para identificar patrones
- Límites de escalabilidad

Estado del arte



Sistema inteligente de salud ocupacional portátil de operación eléctrica basado en Machine Learning



Wearable de control basado en el modelo SVM

- SVM (Support Vector Machine)
- Aprendizaje automático
- Fatiga de operadores
- Signos vitales
- ECG (Electrocardiograma)

Base de datos

Espirometría y audiometría

4233 filas

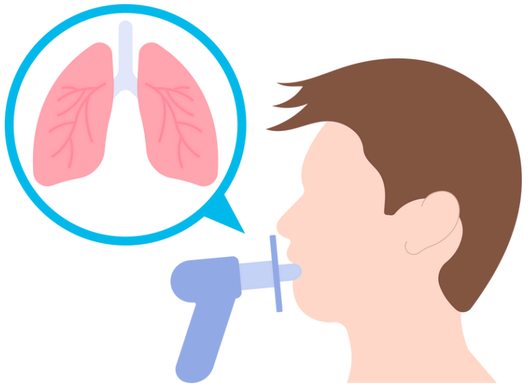
105 columnas

Información demográfica y laboral

- sol_id,
- loc_id
- Edad
- Sexo
- Fecha de nacimiento
- cod2
- Puesto de trabajo
- Ciudad

Pruebas de audiometría

Comentarios y aptitud



Factores de riesgo

- espiro_ante_medicos
- espiro_fumador
- espiro_grupo_etnico

Pruebas de espirometria

- espiro_fvc
- espiro_fev
- espiro_fev_fvc
- espiro_fef
- etc

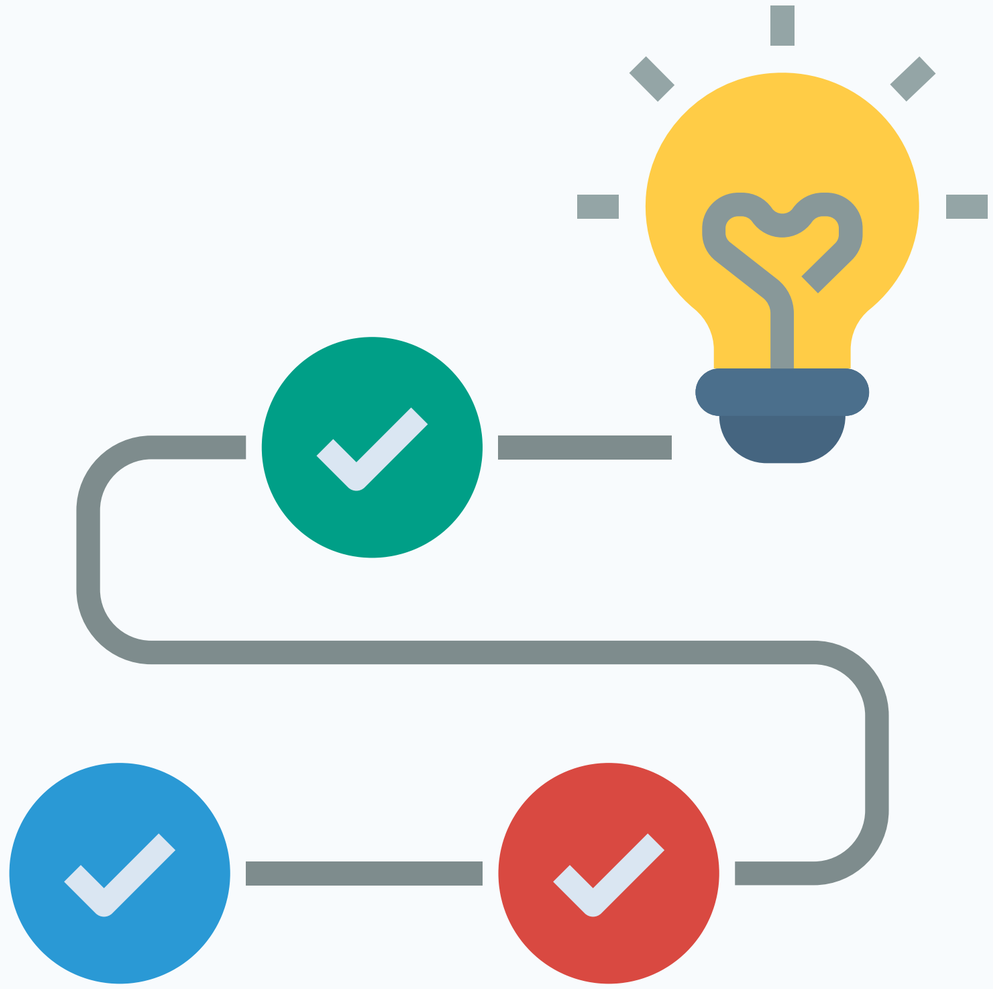
Variables cualitativas

- Sexo
- Puesto de trabajo
- Ciudad
- Audiometría
- Audiometria Otros
- Espiro_ante_medicos
- Espiro_fumador
- Espiro_grupo_etnico
- Espiro_comentario
- Aptitud
- Espirometría

Variables cuantitativas

- Sol_id
- Loc_id
- Edad
- Fecha de nacimiento
- Cod2
- Fecha
- Valores de pruebas de audiometría
- valores de pruebas de espirometría

Metodología



1. Análisis exploratorio de los datos
2. Definición del modelo
3. Mejores hiperparámetros con GridSearchCV
4. Test del modelo

Audiometría

1. Análisis Exploratorio

Manejo de datos

- Se eliminan columnas irrelevantes.
- Se eliminan columnas con exceso de datos nulos.
- Se hace una limpieza de outliers

Categorización de columnas

- GPT 4.0: “Puesto de trabajo”, “Audiometría”
- Manual: “aptitud” , solo dos valores posibles: “apto”, “Observado”

```
print(len(puesto_de_trabajo))
```

```
7454
```

```
len(Audiometria)
```

```
4459
```

Encoding

- Se usó label encoding para “sexo” y “aptitud”
- Se usó frequency encoding para “Puesto de trabajo”, “Audiometría”

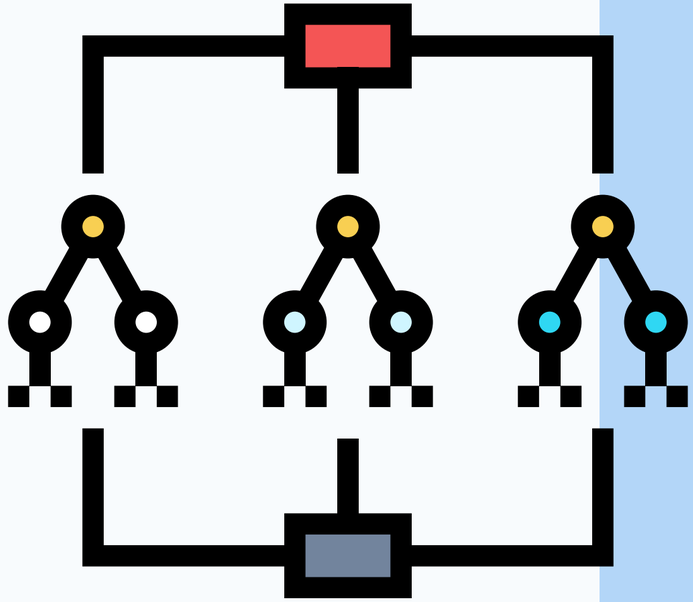
```
<class 'pandas.core.frame.DataFrame'>
Index: 12292 entries, 17 to 39504
Data columns (total 22 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Edad                       12292 non-null  int64
 1   OD Aerea 125               12292 non-null  float64
 2   OD Aerea 250               12292 non-null  float64
 3   OD Aerea 500               12292 non-null  float64
 4   OD Aerea 1000              12292 non-null  float64
 5   OD Aerea 2000              12292 non-null  float64
 6   OD Aerea 3000              12292 non-null  float64
 7   OD Aerea 4000              12292 non-null  float64
 8   OD Aerea 6000              12292 non-null  float64
 9   OD Aerea 8000              12292 non-null  float64
 10  OI Aerea 250               12292 non-null  float64
 11  OI Aerea 500               12292 non-null  float64
 12  OI Aerea 1000              12292 non-null  float64
 13  OI Aerea 2000              12292 non-null  float64
 14  OI Aerea 3000              12292 non-null  float64
 15  OI Aerea 4000              12292 non-null  float64
 16  OI Aerea 6000              12292 non-null  float64
 17  OI Aerea 8000              12292 non-null  float64
 18  aptitud                    12292 non-null  int64
 19  sexo_encoded               12292 non-null  int64
 20  Audiometría_encoded        12292 non-null  float64
 21  Puesto_de_trabajo_encoded  12292 non-null  float64
dtypes: float64(19), int64(3)
memory usage: 2.2 MB
```

2. Definición del modelo

```
LogisticRegression Precisión: 0.5591703944692965
DecisionTreeClassifier Precisión: 0.524603497356649
SVC Precisión: 0.5583570557137048
KNeighborsClassifier Precisión: 0.5510370069133794
RandomForestClassifier Precisión: 0.5766571777145181
```

Random Forest Classifier

- Robustez
- Manejo de datos desbalanceados
- Generalización



3. Tuneo de hiperparámetros

GridSearchCV

```
'n_estimators': [50, 100, 200],
'max_depth': [None, 10, 20],
'min_samples_split': [2, 5, 10],
'min_samples_leaf': [1, 2, 4]
```

Mejores hiperparámetros:

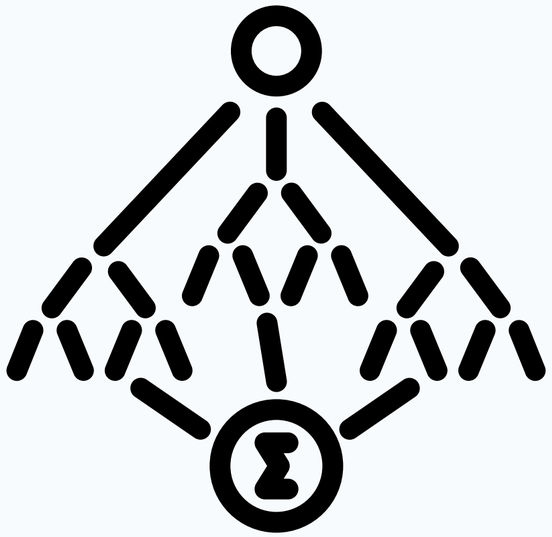
- 'max_depth': 2
- 'min_samples_leaf: 4
- 'min_samples_split': 2
- 'n_estimators': 50

4. Training y testing

```
# Divide los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Proporción 20% testing, y 80% de training

Validación cruzada



1. Análisis exploratorio de los datos
2. Definición del modelo
3. Ajuste de hiperparámetros
4. Train y test del modelo

Espirometría

1. Análisis Exploratorio

Evaluación de columnas

- Selección de columnas relevantes
- Eliminación de columnas innecesarias
- Consideración de columnas que aporten al examen espirométrico

Detección de Nan, vacios y valores atípicos

- Eliminación de valores nan y vacíos
- Filtración de valores atípicos con el rango intercuartilico.
- Eliminacion de valores atípicos

```
df_espiro_filtrado = df_espiro_filtrado.dropna()
```

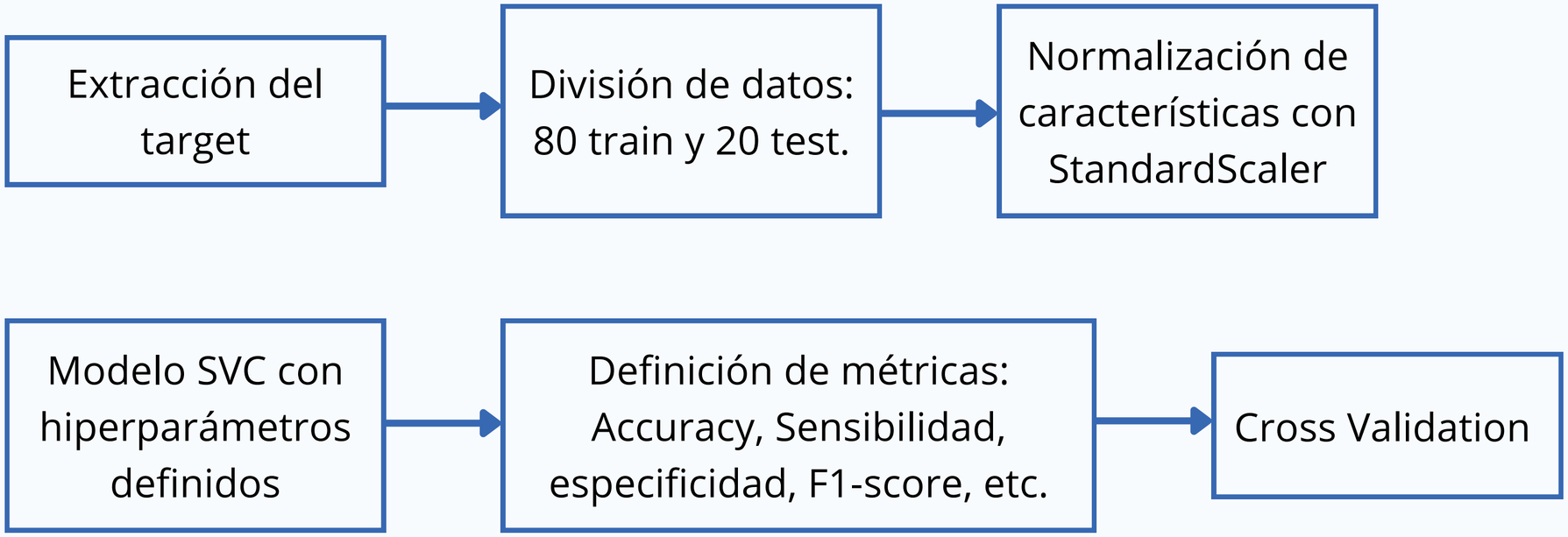
Reemplazo de valores

- Valores de texto a binario: Fumador, sexo, aptitud.
- Agrupamiento de aptitud a dos únicos tipos de valores

```
aptitud
Apto con Restricciones    4819
Apto                      4499
Name: count, dtype: int64
```

2. Definición del modelo

Modelo de clasificación binaria




```
model = SVC(kernel='linear', C=1, random_state=42)
```

3. Ajuste de hiperparámetros

```
C_value = 10
kernel_type = 'poly'
gamma_value = 'auto'
```

4. Train y test

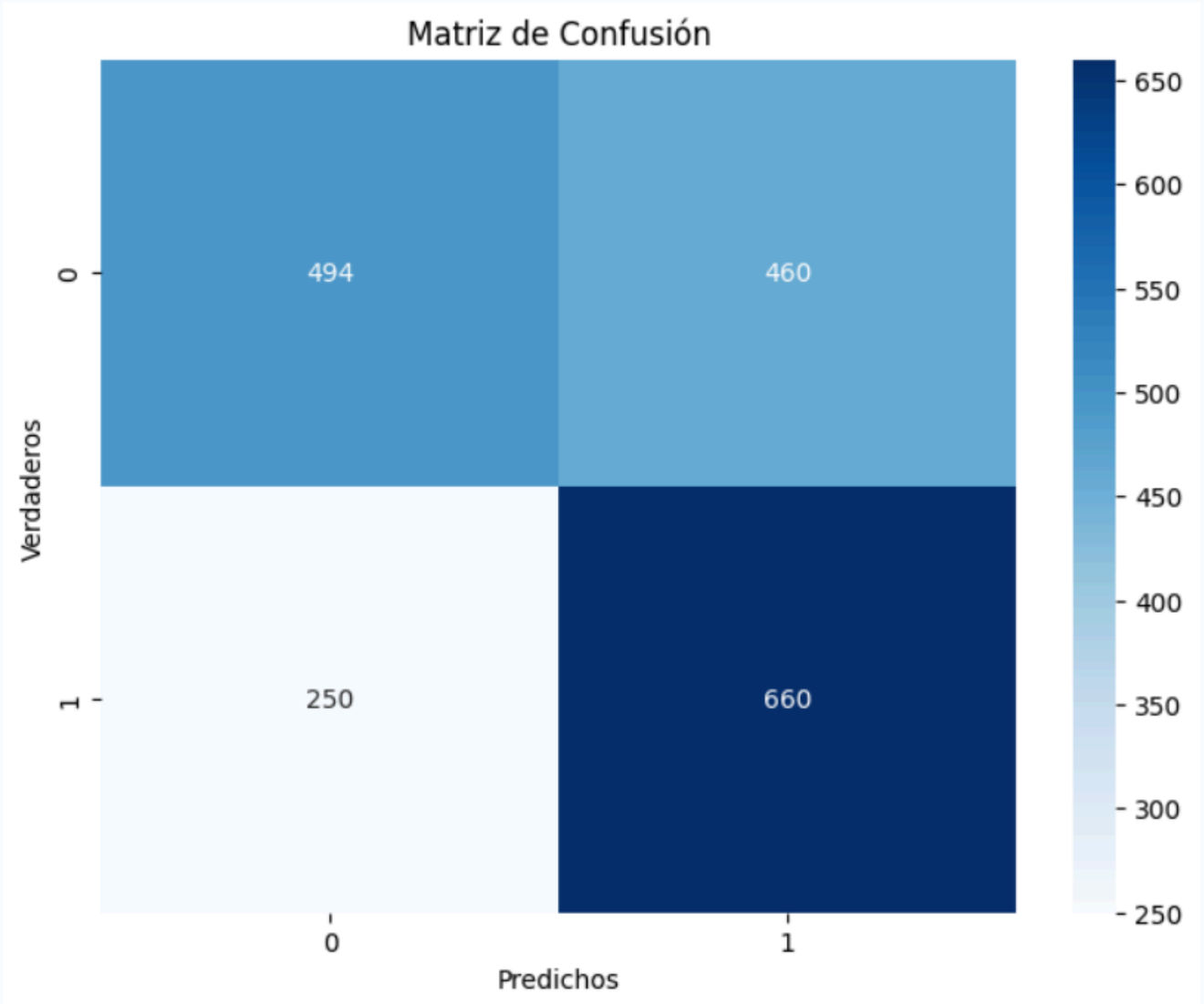
```
# Evaluación utilizando validación cruzada
scores = cross_val_score(model, X_train, y_train, cv=5)
print(f'Cross-validation accuracy scores: {scores}')
print(f'Mean cross-validation accuracy: {scores.mean()}')
```

Resultados

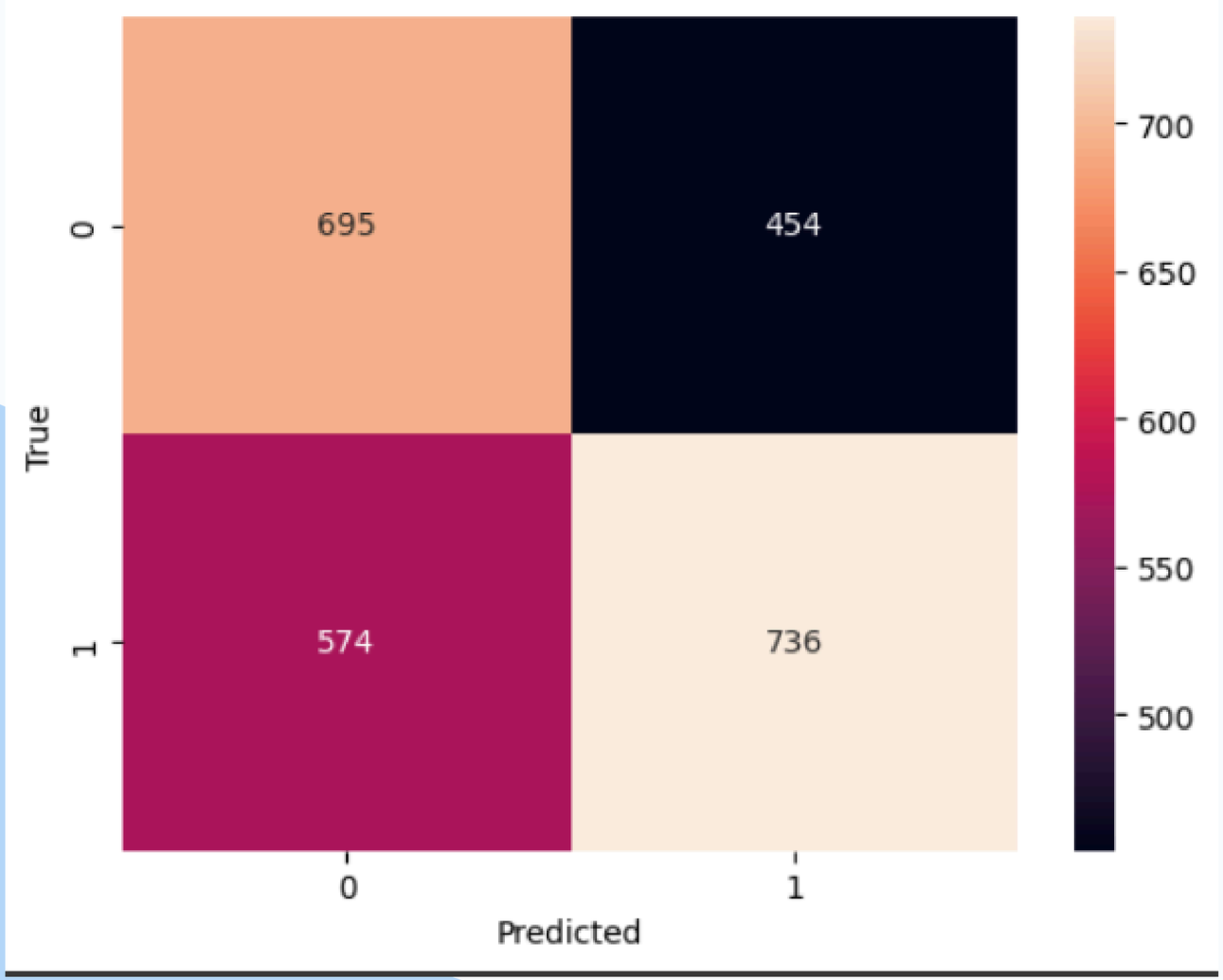
Espirometría

```
Métricas de rendimiento:
     Metric    Value
   Accuracy 0.619099
Sensitivity 0.725275
Specificity 0.517820
   F1 Score 0.615247

Resultados de Validación Cruzada:
 Fold    Score
    1 0.623072
    2 0.621060
    3 0.610999
    4 0.583501
    5 0.613423
Mean CV Score: 0.6104
```



Audiometría




```
 Sensitivity (Recall): 0.5618320610687023
 Specificity: 0.6048738033072236
Accuracy: 0.5819438796258641
Precision: 0.6184873949579832
Recall: 0.5618320610687023
F1-score: 0.5888
```

Conclusiones

Muchas Gracias

Por ver esta presentación