

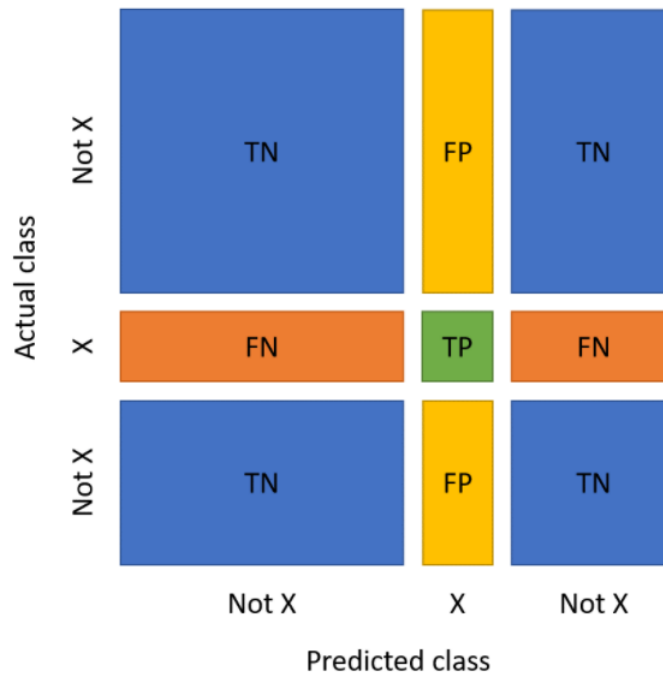
## INVESTIGAR LAS CLASIFICACIONES ERRÓNEAS

### Identificar las Clasificaciones Erróneas más Comunes

#### *La matriz de confusión*

Para cualquier clase de respuesta X, puede dividir las predicciones de un modelo de machine learning en cuatro grupos:

- **Positivos verdaderos** (verde): se predice que es X y en realidad es X
- **Negativos verdaderos** (azul): se predice que no es X y en realidad no es X
- **Falsos positivos** (amarillo): se predice que es X pero en realidad no es X
- **Falsos negativos** (naranja): se predice que no es X pero en realidad es X



#### Actividad 1

Cuando se crea una gráfica de confusión, se puede agregar información sobre la tasa de falsos negativos y falsos positivos de cada clase agregando resúmenes de filas o columnas, respectivamente.

```
confusionchart(...,"RowSummary","row-normalized");
```

**Tarea:** Cargue los datos, un modelo knn entrenado y muestre la matriz de confusión:

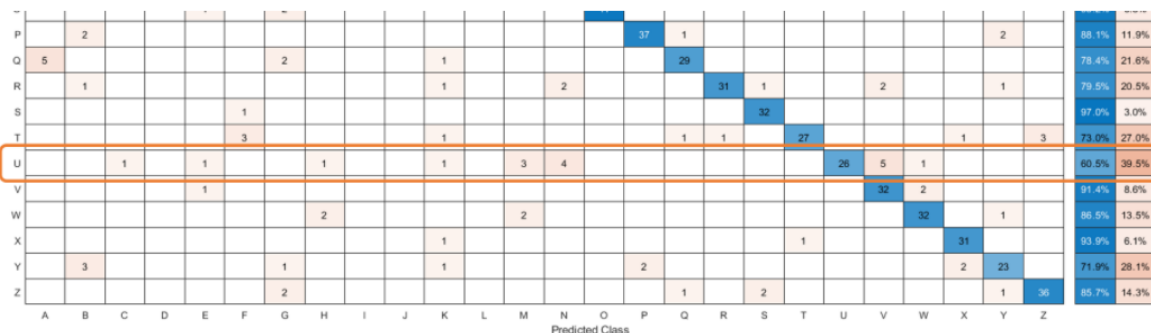
```
load datosletras.mat
load modelo_pred.mat
confusionchart(datos_prueba.Character,letra_pred)
```

Vuelva a crear la gráfica de confusión con la información resumida de filas normalizada.

## Falsos negativos

El resumen de filas muestra la tasa de falsos negativos para cada clase. (Con 26 letras, tendrá que ampliar la gráfica para que los valores sean visibles). Esto muestra qué letras el modelo kNN son más difíciles de identificar (es decir, las letras que el modelo más a menudo piensa que son otra). Este modelo tiene una dificultad particular con la letra U, que a menudo se confunde con M, N o V.

Algunas confusiones parecen razonables, como U/V o H/N. Otros son más sorprendentes, como U/K. Tras haber identificado las clasificaciones erróneas de interés, probablemente querrá ver algunas muestras de datos específicos para entender qué está causando la clasificación errónea.



## Actividad 2

Puede usar operadores relacionales y lógicos (como `==`, `~`, `&` y `|`) para identificar las observaciones que se deben estudiar más a fondo.

**Tarea:** Utilice operadores relacionales y lógicos para crear un arreglo lógico llamado `falsos_neg` que identifica los casos de los datos de prueba en los que la letra U se ha clasificado como otra. Es decir, elementos en los que la clase verdadera (`datos_prueba.Character`) es "U" y la clase predicha (`letra_pred`) no era "U".

## Actividad 3

Recuerde que la propiedad `Files` de un almacén de datos contiene los nombres de archivo de los datos originales. Por lo tanto, cuando se importan los datos y se extraen las características, se puede mantener un registro de qué archivo de datos está asociado con cada observación. El arreglo de cadenas `archivos_prueba` contiene los nombres de los archivos de los datos de prueba.

**Tarea:** Utilice el arreglo lógico `falsos_neg` como índice en `archivos_prueba` para determinar los nombres de los archivos de las observaciones que se han clasificado incorrectamente como la letra U. Almacene el resultado en una variable llamada `archivos_falsos_neg`.

Del mismo modo, utilice `falsos_neg` como índice en `letra_pred` para determinar las letras predichas asociadas. Almacene el resultado en una variable llamada `pred_falsos_neg`.

#### Actividad 4

El cuarto elemento de `pred_falsos_neg` es N, que es una clasificación errónea común para la letra U. ¿Qué aspecto tiene esta muestra en particular?

**Tarea:** Utilice `readtable` para importar los datos del cuarto elemento de `archivos_falsos_neg` en una tabla llamada `U_ma1`. Visualice la letra mediante la representación de Y frente a X.

#### Tarea adicional

Piense en la posición del lápiz a través del tiempo para esta U y para una N típica. ¿Es razonable que puedan confundirse entre sí?

Observe algunas de las otras clasificaciones erróneas. Puede agregar un título a la gráfica para mostrar lo que el modelo ha predicho:

```
title("Prediccion: "+string(pred_falsos_neg(4)))
```

#### Archivos requeridos:

`datosletras.mat`

`predmodel.mat`

`user024_U_1.txt`