Reclamos por Obras Públicas en CABA

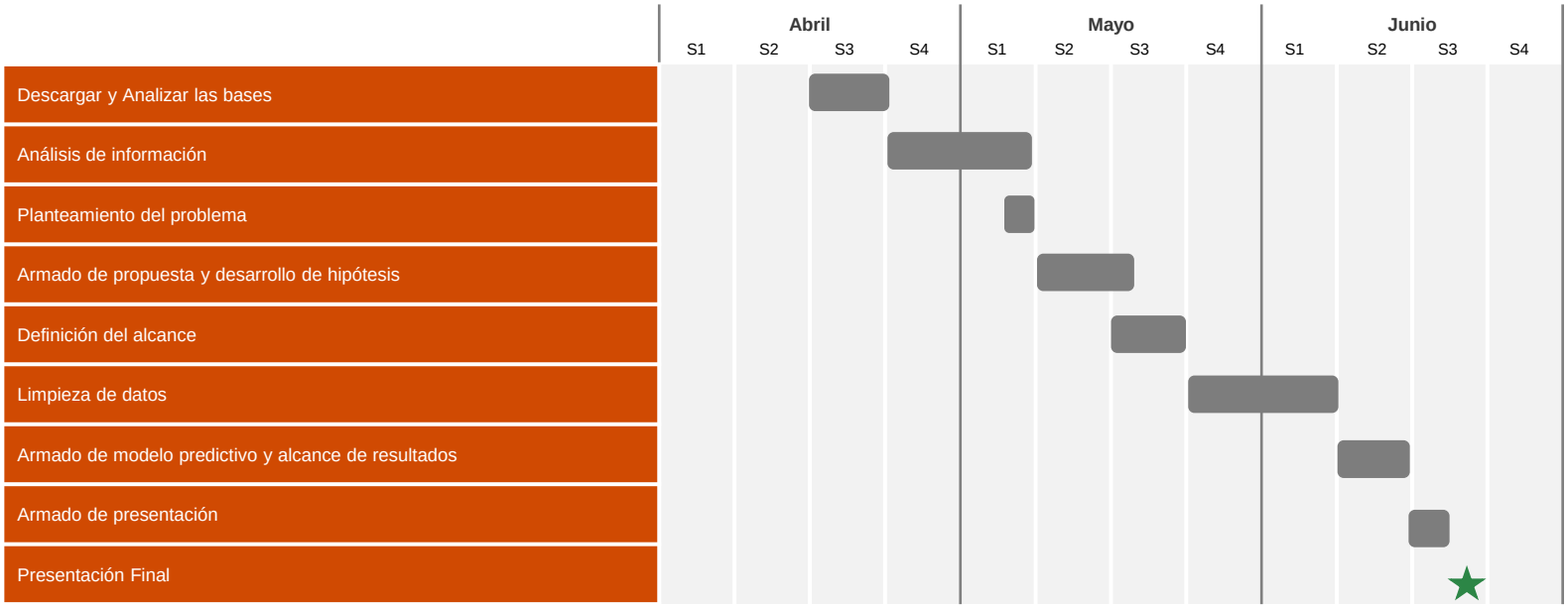
Junio 2019

Agenda

1

Cronograma

Cronograma



No hubo desvíos significativos entre los tiempos planificados en el proyecto y los tiempos de ejecución del mismo.

EANT

Buenos Aires Ciudad

2

Nuestra Propuesta

Nuestra propuesta

¿En qué medida aumentan los reclamos ante el inicio de obras en la vía pública?

La propuesta es poder responder la pregunta a partir de la generación de un modelo predictivo que trabaje sobre una variable objetivo llamada "**porcentaje de aumento de reclamos ante inicio de obras**"

La Variable

$$\frac{\text{Cantidad de reclamos en un lapso de tiempo X, en un radio de Y cuadras}}{\text{Promedio de reclamos en el mismo lapso de tiempo, en dicho radio de cuadras}} \%$$

El Modelo Predictivo

- Una vez obtenido el porcentaje de aumento para cada obra iniciada que figura en el dataset, se entrena un modelo predictivo.
- Las variables independientes a incluir se definirían en el propio proceso, pero, presumiblemente, la latitud y la longitud serán centrales.
- La variable objetivo sería el mencionado porcentaje de aumento.
- El tipo de modelo y sus respectivos hiperparámetros se definirían, también, a partir de la constatación práctica de su precisión.

EANT

Buenos Aires Ciudad

3

Algunas estadísticas

Algunas estadísticas

Durante el período 2017-2019 (comprendido por el primer trimestre del corriente año) se realizaron alrededor de 487 obras y se contabilizaron 2266 reclamos aproximadamente.
Dentro del total de reclamos el 20% fueron realizados por la Comuna 1, seguido por la Comuna 14, que representó el 14% de los reclamos.

Base de Reclamos y Denuncias

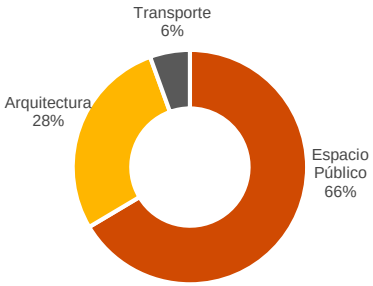
Base de Obras

2266
Reclamos

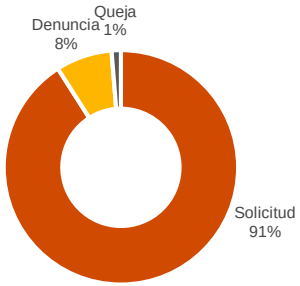
487
Obras

2,3
Años

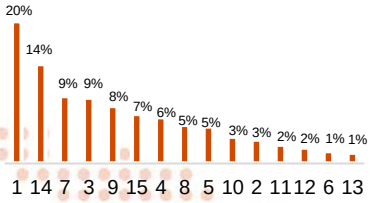
Tipo de Obra



Tipo de Reclamo



Reclamos por Comuna



2017-2019
Período

Comuna 1 = San Nicolás, Puerto Madero, San Telmo, Montserrat y Constitución.

Comuna 14 = Palermo

EANT

Buenos Aires Ciudad

4

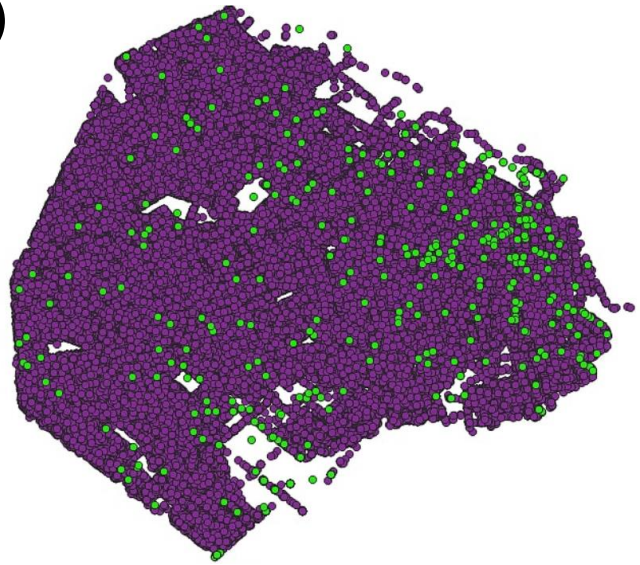
Geolocalización

Geolocalización

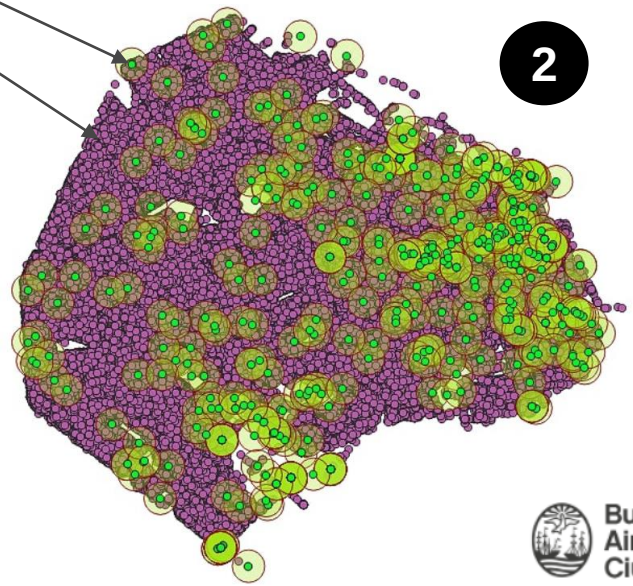
QGIS es un Sistema de Información Geográfica de código abierto.

Se observa la cantidad de obras (puntos verdes) y los reclamos (puntos violetas) asociados y no asociados a cada obra

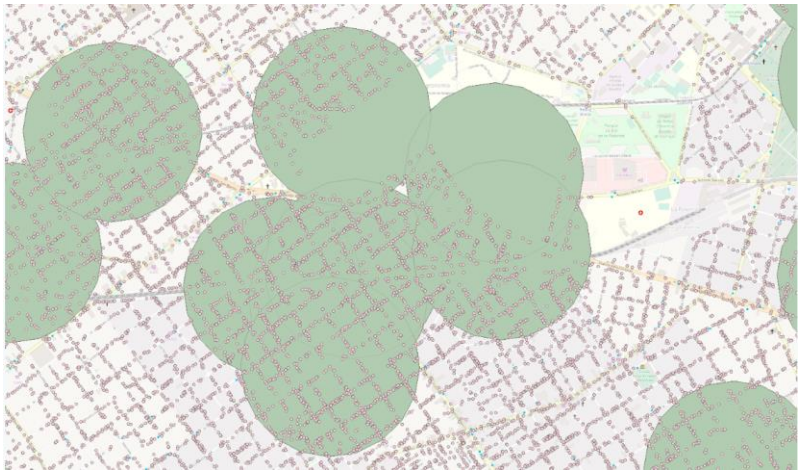
1



2

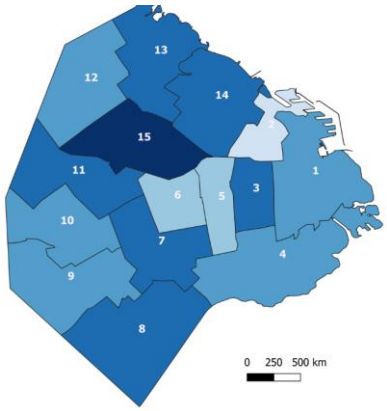


3



Geolocalización

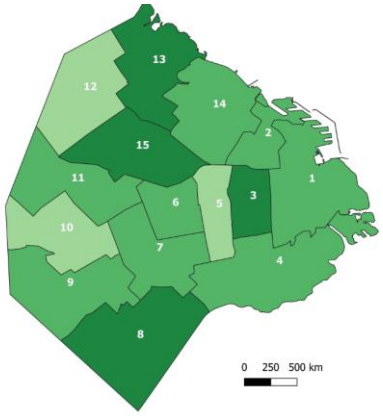
2017



Ranking de reclamos y denuncias por Comuna

- *> 9000 = Comuna: 15*
- *(8001 – 9000) = Comunas: 15, 8, 7, 11, 13, 14, 3*
- *(7001 – 8000) = Comunas: 10, 9, 4, 1, 12*
- *(6001 – 7000) = Comunas: 5, 6*
- *(5001 – 6000) = Comuna: 2*

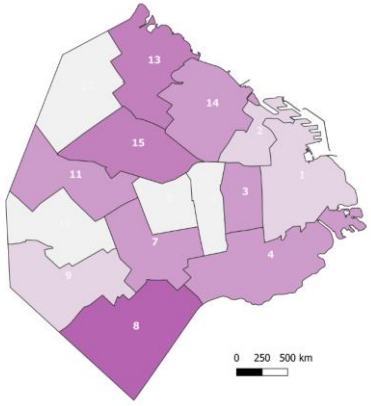
2018



Ranking de reclamos y denuncias por Comuna

- *> 9000 = 0*
- *(8001 – 9000) = Comunas: 15, 8, 13, 3*
- *(7001 – 8000) = Comunas: 9, 11, 6, 7, 4, 1, 2, 14,*
- *(6001 – 7000) = Comunas: 12, 10, 5*
- *(5001 – 6000) = 0*

IT 2019



Ranking de reclamos y denuncias por Comuna

- *> 9000 = 8*
- *(8001 – 9000) = Comunas: 15, 13*
- *(7001 – 8000) = Comunas: 14, 3, 4, 7, 11*
- *(6001 – 7000) = Comunas: 9, 2, 1*
- *(5001 – 6000) = Comunas: 5, 10, 12, 6*

5

El Modelo Predictivo

El Modelo Predictivo

Árbol de Decisión

```
from sklearn.preprocessing import StandardScaler
from sklearn.tree import DecisionTreeRegressor
from sklearn.model_selection import cross_val_score
y_pred = regressor.predict(X_test)
from sklearn.metrics import mean_squared_error
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
print(np.round(rmse, 2))
```

```
6.17
```

RMSE = 6,17

Regresión Lineal

```
from sklearn.preprocessing import StandardScaler
from sklearn import linear_model
from sklearn.metrics import mean_squared_error
y_pred = regressor.predict(X_test)
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
print(np.round(rmse, 2))
```

```
7.1
```

RMSE = 7,1

EANT

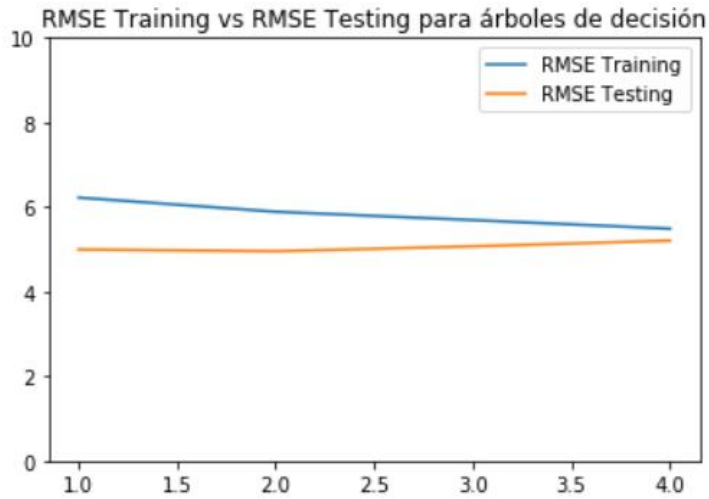
Buenos Aires Ciudad

El Modelo Predictivo

Árbol de Decisión

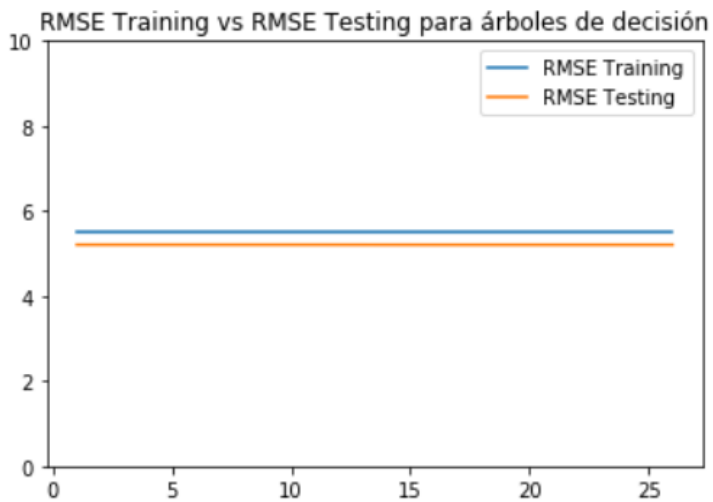
```
rmses_train = []
rmses_test = []
for max_depth in range(1,5, 1):
    regressor = DecisionTreeRegressor(random_state=0, max_depth=max_depth)
    regressor.fit(X_train, y_train)
    y_pred_train = regressor.predict(X_train)
    y_
    rm
    rm
    rm
    rm
```

```
import matplotlib.pyplot as plt
%matplotlib inline
plt.plot(range(1,5, 1), rmses_train, label='RMSE Training')
plt.plot(range(1,5, 1), rmses_test, label='RMSE Testing')
plt.ylim((0, 10))
plt.legend(loc="best")
plt.title("RMSE Training vs RMSE Testing para árboles de decisión")
plt.show()
```




```
rmses_train = []
rmses_test = []
for random_state in range(1,30, 5):
    regressor = DecisionTreeRegressor(random_state=random_state, max_depth=4)
    regressor.fit(X_train, y_train)
    y_pred_train = regressor.predict(X_train)
    y_
    rm
    rm
    rm
    rm
```

```
import matplotlib.pyplot as plt
%matplotlib inline
plt.plot(range(1,30, 5), rmses_train, label='RMSE Training')
plt.plot(range(1,30, 5), rmses_test, label='RMSE Testing')
plt.ylim((0, 10))
plt.legend(loc="best")
plt.title("RMSE Training vs RMSE Testing para árboles de decisión")
plt.show()
```



EANT

Buenos Aires Ciudad

El Modelo Predictivo

Árbol de Decisión

```
def nmsq2rmse(score):
    regressor = DecisionTreeRegressor(random_state=0, max_depth=4)
        regressor.fit(X_train, y_train)
        y_pred = regressor.predict(X_test)
predicciones = pd.concat([val_real.rename('Valor real'),
                          val_pred.rename('Valor Pred') ,abs(val_real-val_pred).rename('Dif(+/-)')] ,  axis=1)
```

	Valor real	Valor Pred	Dif(+/-)
0	13.000	0.000	13.000
1	2.000	6.044	4.044
2	0.000	0.000	0.000
3	9.000	6.044	2.956
4	0.000	6.044	6.044
5	0.000	0.783	0.783
6	0.000	0.783	0.783
7	6.000	6.044	0.044
8	8.000	6.044	1.956
9	13.000	6.044	6.956
10	6.000	6.044	0.044
11	3.000	1.067	1.933
12	7.000	6.044	0.956
13	3.000	0.783	2.217
14	17.000	0.062	16.938
15	0.000	1.067	1.067
16	2.000	0.783	1.217
17	1.000	6.044	5.044
18	0.000	1.067	1.067
19	9.000	6.500	2.500
20	0.000	6.044	6.044
21	0.000	6.044	6.044
22	20.000	6.044	13.956
23	9.000	6.044	2.956
24	1.000	1.067	0.067
25	7.000	6.044	0.956
26	0.000	0.783	0.783
27	4.000	6.044	2.044
28	6.000	6.044	0.044
29	2.000	1.000	1.000

El max_depth que mejor funciona es 4. Sin embargo, las diferencias entre el valor real y el valor predictivo son muy altas.

Habría que continuar mejorando el modelo.

EANT

Buenos Aires Ciudad

El Modelo Predictivo

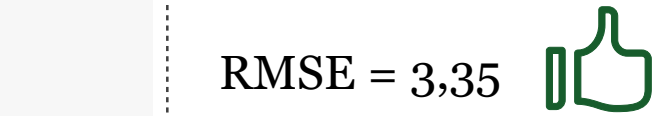
Ridge Regression

```
import numpy as np
np.random.seed(123)
from sklearn.preprocessing import StandardScaler
X =  from sklearn.linear_model import Ridge
y =
X_tr clf = Ridge(alpha=1.0)
```

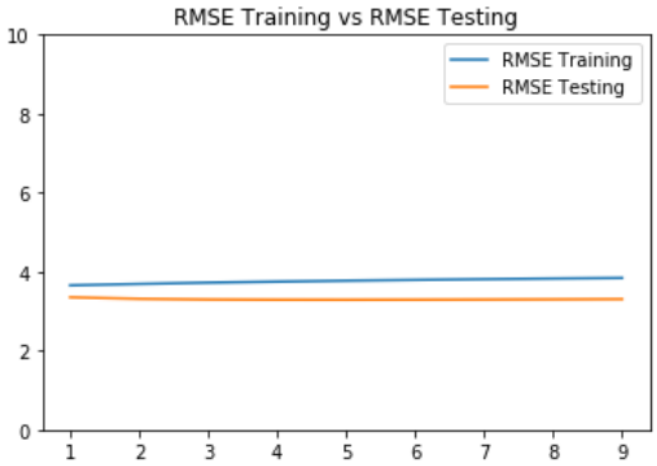
```
y_pred = clf.predict(X_test)
from sklearn.metrics import mean_squared_error
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
print(np.round(rmse, 2))
```

```
3.35
```

RMSE = 3,35



```
rmses_train = []
rmses_test = []
for alpha in range(1,10, 1):
    regressor = Ridge(alpha=alpha, max_iter=1000)
    regressor.fit(X_train, y_train)
    import matplotlib.pyplot as plt
    %matplotlib inline
    plt.plot(range(1,10, 1), rmses_train, label='RMSE Training')
    plt.plot(range(1,10, 1), rmses_test, label='RMSE Testing')
    plt.ylim((0, 10))
    plt.legend(loc="best")
    plt.title("RMSE Training vs RMSE Testing")
    plt.show()
```



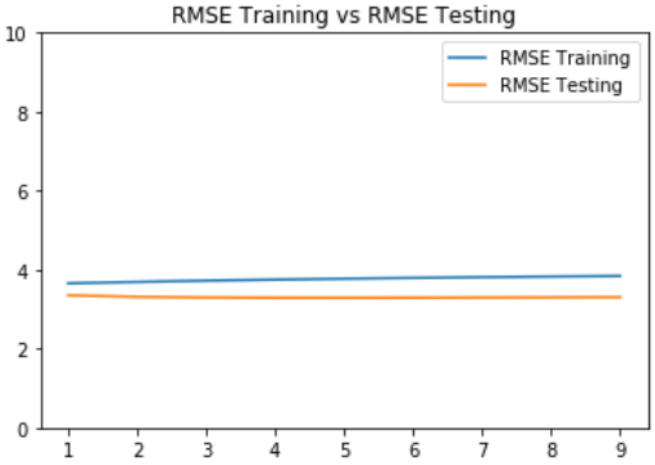
EANT

Buenos Aires Ciudad

El Modelo Predictivo

Ridge Regression

```
rmses_train = []
rmses_test = []
for alpha in range(1,10, 1):
    regressor = Ridge(alpha=alpha, max_iter=1000)
    regressor.fit(X_train, y_train)
    import matplotlib.pyplot as plt
    %matplotlib inline
    plt.plot(range(1,10, 1), rmses_train, label='RMSE Training')
    plt.plot(range(1,10, 1), rmses_test, label='RMSE Testing')
    plt.ylim((0, 10))
    plt.legend(loc="best")
    plt.title("RMSE Training vs RMSE Testing")
    plt.show()
```



	Valor real	Valor Pred	Dif(+/-)
0	3.0	4.390546	1.390546
1	0.0	4.728464	4.728464
2	0.0	3.671373	3.671373
3	2.0	4.949766	2.949766
4	0.0	1.542686	1.542686
5	9.0	4.805227	4.194773
6	0.0	0.306757	0.306757
7	0.0	3.669372	3.669372
8	7.0	4.533507	2.466493
9	0.0	1.365921	1.365921
10	0.0	1.673469	1.673469
11	9.0	4.387521	4.612479
12	5.0	4.544801	0.455199
13	0.0	-0.085498	0.085498
14	0.0	2.073234	2.073234
15	2.0	4.636757	2.636757

Luego de sacar los outliers y aplicar el modelo de Ridge Regression, el modelo mejoró significativamente.
Las diferencias entre el valor real y el valor predictivo, bajaron.

Se podría seguir trabajando en mejorar el modelo.

Buenos Aires Ciudad

6

Conclusiones

Conclusiones

¿En qué medida aumentan los reclamos ante el inicio de obras en la vía pública?

"Según los modelos utilizados podemos afirmar que alcanzamos un acercamiento parcial al reconocimiento del aumento de los reclamos"

Soluciones para mejorar el modelo: al quitar los outliers el modelo mejoró significativamente. De todas maneras, se podría continuar mejorando el modelo al agregar o sacar atributos, y/o modificarlos (por ejemplo: aumentando el tiempo de 7 días para considerar una mayor cantidad de reclamos o aumentando el radio para detectar más reclamos en cada obra)

Issues y Mejoras

- A medida que avanzamos con el trabajo, surgieron problemáticas que fuimos resolviendo.
- El 80% del trabajo estuvo destinado a la limpieza de los datos.
- Ante el armado del modelo predictivo, advertimos que no contemplamos las obras que no tenías reclamos, pudiendo caer en Overfitting.
- Acotamos el universo a estudiar a fin de mejorar el análisis.
- La diferencia de criterios en los campos de las bases anuales representaron un problema al momento de realizer la unificación de las bases.

EANT

Buenos Aires Ciudad

¡Gracias!

EANT

Buenos Aires Ciudad