Projet Jedha

Les déterminants de la réussite scolaire des élèves en Afrique : cas du Kenya

Abou & Boris

Enseignant : Badr Ouali

2020

Contexte de l'étude

- Le SACMEQ est un consortium qui regroupe 15 pays d'Afrique australe et orientale qui permet :
  - Repenser les politiques , d'orienter les réformes, d'élaborer des programmes éducatifs,
  - Gérer les défis lancés touchant la qualité de l'enseignement ...

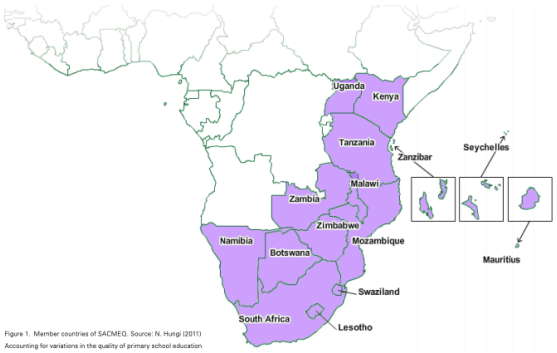


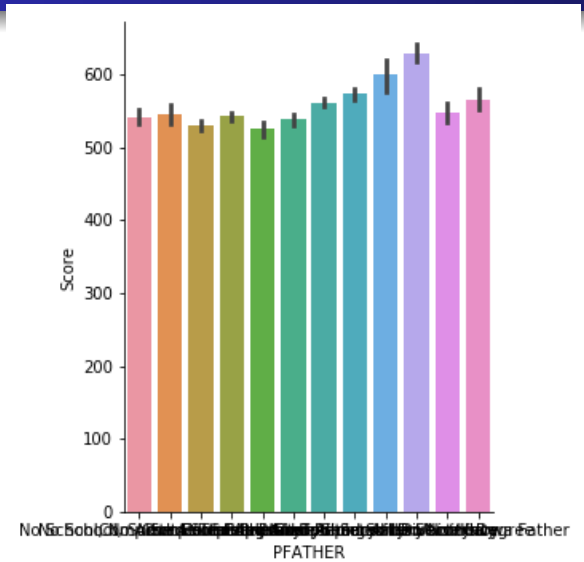
Figure 1. Member countries of SACMEQ. Source: N. Hungi (2011) Accounting for variations in the quality of primary school education

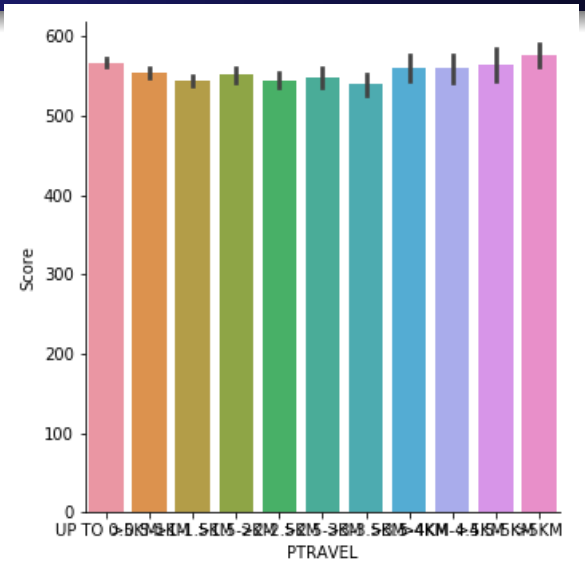
Problématique

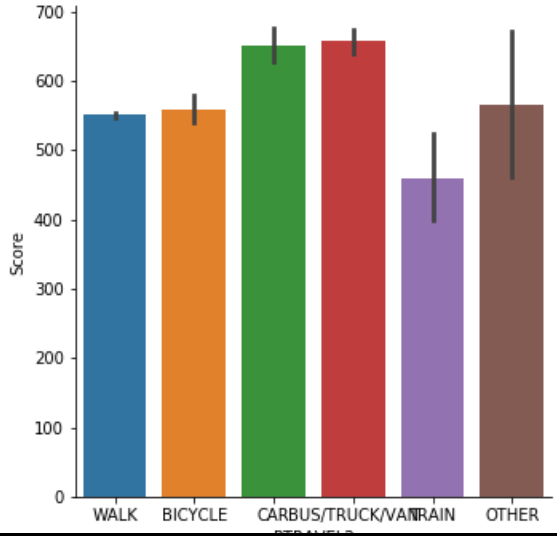
- Les inégalités scolaires :
  - Isolement des écoles
  - Régionales, rurales / urbaines, isolées / non isolées,
  - Entre les filles et garçons,
  - Entre établissements publics/privés.

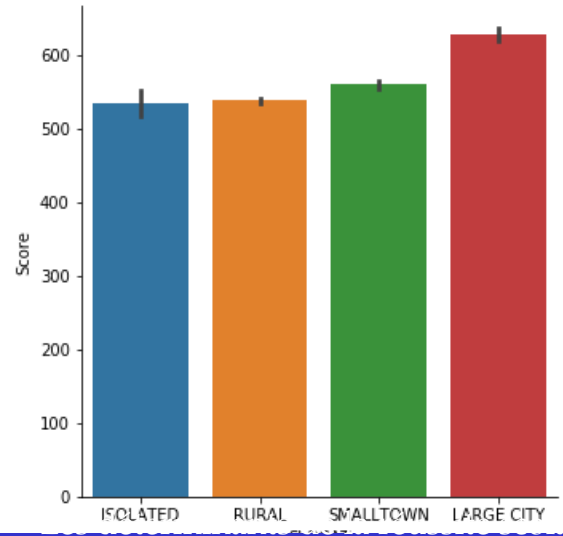
Plan

Data Exploration









Data Preparation

Les variables

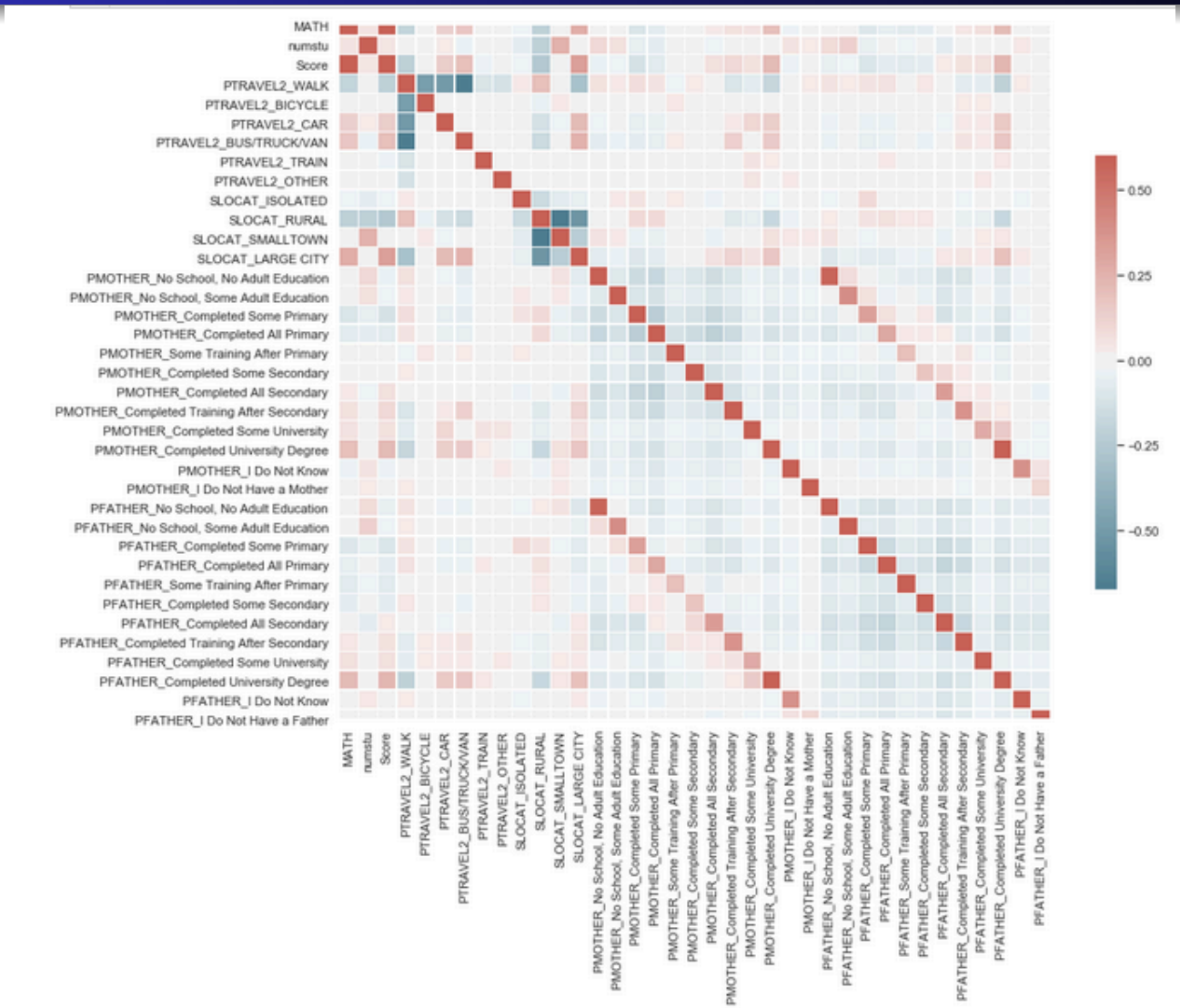
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 4436 entries, 0 to 4435
Data columns (total 55 columns):
ID                  4436 non-null float32
COUNTRY             4436 non-null object
REGION              4436 non-null object
DISTRICT            4436 non-null object
SCHOOL              4436 non-null float64
PUPIL               4436 non-null float64
READ                4436 non-null float64
MATH                4433 non-null float64
read_prof           4191 non-null float64
math_prof           4071 non-null float64
numstu              4436 non-null float64
PSEX                4436 non-null category
PTRAVEL             4436 non-null category
PTRAVEL2            4436 non-null category
PMOTHER             4436 non-null category
PFATHER             4422 non-null category
STYPE               4436 non-null category
```

Data Preparation

Les variables

- Nettoyages de la base de données
- Calcul de la moyenne de la variable à prédire
  $score = (READ + MATH)/2$
- Suppression des outiliers des et des variables avec trop de valeurs manquantes
- Créations des dummies

Correlation



Variables selectionnées

Out[298]:

	Features	Value
12	SLOCAT_LARGE CITY	0.463633
0	DISTANCE	0.102244
34	PFATHER_Completed University Degree	0.085936
22	PMOTHER_Completed University Degree	0.058412
3	PTRAVEL2_WALK	0.045986
19	PMOTHER_Completed All Secondary	0.038504
11	SLOCAT_SMALLTOWN	0.020682
1	PSEX_BOY	0.018845
6	PTRAVEL2_BUS/TRUCK/VAN	0.018127
10	SLOCAT_RURAL	0.017175
2	PSEX_GIRL	0.014631
13	PMOTHER_No School, No Adult Education	0.011365
33	PFATHER_Completed Some University	0.010624
20	PMOTHER_Completed Training After Secondary	0.009698
14	PMOTHER_No School, Some Adult Education	0.009318
23	PMOTHER_I Do Not Know	0.008513
32	PFATHER_Completed Training After Secondary	0.008176
15	PMOTHER_Completed Some Primary	0.007083

Modèle
Decision Tree

[49]:
```
# Evaluation du modèle
print("Train Score : {}".format(dtree.score(X_train, y_train)))
print('Test Score : {}'.format(dtree.score(X_test, y_test)))
```

```
Train Score : 0.058833069127457005
Test Score : 0.09934970520614084
```

[50]:
```
y_pred=dtree.predict(X_test)
```

[51]:
```
from sklearn import metrics

print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
```

```
Mean Absolute Error: 68.78260780647231
Mean Squared Error: 7876.010218259053
Root Mean Squared Error: 88.74688849902881
```

Modèle
Random Forest 1

53]:
```
# Evaluation du modèle
print("Train Score : {}".format(regressor.score(X_train, y_train)))
print('Test Score : {}'.format(regressor.score(X_test, y_test)))
```

```
Train Score : 0.26866226842892305
Test Score : 0.20075158611484614
```

54]:
```
from sklearn import metrics

print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
```

```
Mean Absolute Error: 64.94272155326026
Mean Squared Error: 6989.270653742013
Root Mean Squared Error: 83.60185795627997
```

Modèle
Random Forest 2

]:
```
# Tentons une random forest
from sklearn.ensemble import RandomForestRegressor

regressor = RandomForestRegressor(n_estimators = 1000, random_state=0)
regressor.fit(X_train, y_train)
y_pred = regressor.predict(X_test)
```

]:
```
# Evaluation du modèle
print("Train Score : {}".format(regressor.score(X_train, y_train)))
print('Test Score : {}'.format(regressor.score(X_test, y_test)))
```

```
Train Score : 0.5885003430235788
Test Score : 0.09541805193929653
```

]:
```
from sklearn import metrics

print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
```

```
Mean Absolute Error: 69.17453652480096
Mean Squared Error: 7910.391755114491
Root Mean Squared Error: 88.9403831513812
```

Conclusion

- grande ville,
- distance,
- Niveau d'education du père

Perspective

Modèle à trois niveaux