```
from google.colab import files
uploaded = files.upload()
```

选择文件 big_data.csv

- **big_data.csv**(application/vnd.ms-excel) - 139769656 bytes, last modified: 2020/4/28 - 100% done

```
Saving big_data.csv to big_data.csv
```

```
import pandas as pd
data = pd.read_csv(’big_data.csv’)
pd.DataFrame.from_records(data)
```

	arr_delay	distance	presidents_day	easter	memorial_day	independence_day	labor_d
0	-7	270	0	0	0	0	
1	5	1995	0	0	0	0	
2	7	621	0	0	0	0	
3	-33	2065	0	0	0	0	
4	-19	1771	0	0	0	0	
...	...	...	...	...	...	...	
177507	8	337	0	0	0	0	
177508	10	1024	0	1	0	0	
177509	44	1585	0	1	0	0	
177510	-15	337	0	0	0	0	
177511	-10	624	0	0	0	0	

177512 rows × 382 columns

```
import pandas as pd
from sklearn.model_selection import train_test_split
import xgboost as xgb
from sklearn.metrics import mean_squared_error
import numpy as np

data.dropna(axis=0, subset=[’arr_delay’], inplace=True)

y = data.arr_delay
X = data.drop([’arr_delay’], axis=1).select_dtypes(exclude=[’object’])


X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=123)
```

Mean Absolute Error results here

```
data_dmatrix = xgb.DMatrix(data=X,label=y)


xg_reg = xgb.XGBRegressor(objective =’reg:linear’, colsample_bytree = 0.3, learning_rate = 0.1,
                max_depth = 5, alpha = 10, n_estimators = 10)


xg_reg.fit(X_train,y_train)
```

```
preds = xg_reg.predict(X_test)
```

```
[18:01:39] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor
```

```
rmse = np.sqrt(mean_squared_error(y_test, preds))
print("RMSE: %f" % (rmse))
```

```
RMSE: 31.686198
```

changes for model tuning

```
params = {"objective":"reg:linear",'colsample_bytree': 0.3,'learning_rate': 0.1,
                'max_depth': 5, 'alpha': 10}

cv_results = xgb.cv(dtrain=data_dmatrix, params=params, nfold=3,
                    num_boost_round=50,early_stopping_rounds=10,metrics="rmse", as_pandas=
```

```
[18:01:56] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor
[18:02:01] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor
[18:02:04] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor
```

```
cv_results.head()
```

	train-rmse-mean	train-rmse-std	test-rmse-mean	test-rmse-std
0	41.508718	0.103943	41.521145	0.207252
1	39.282958	1.427370	39.298312	1.723997
2	38.255065	2.349752	38.284343	2.695878
3	38.171123	2.361523	38.239312	2.694978
4	37.278604	3.411559	37.357897	3.752378

```
print((cv_results["test-rmse-mean"]).tail(1))
```

```
49    19.407878
Name: test-rmse-mean, dtype: float64
```

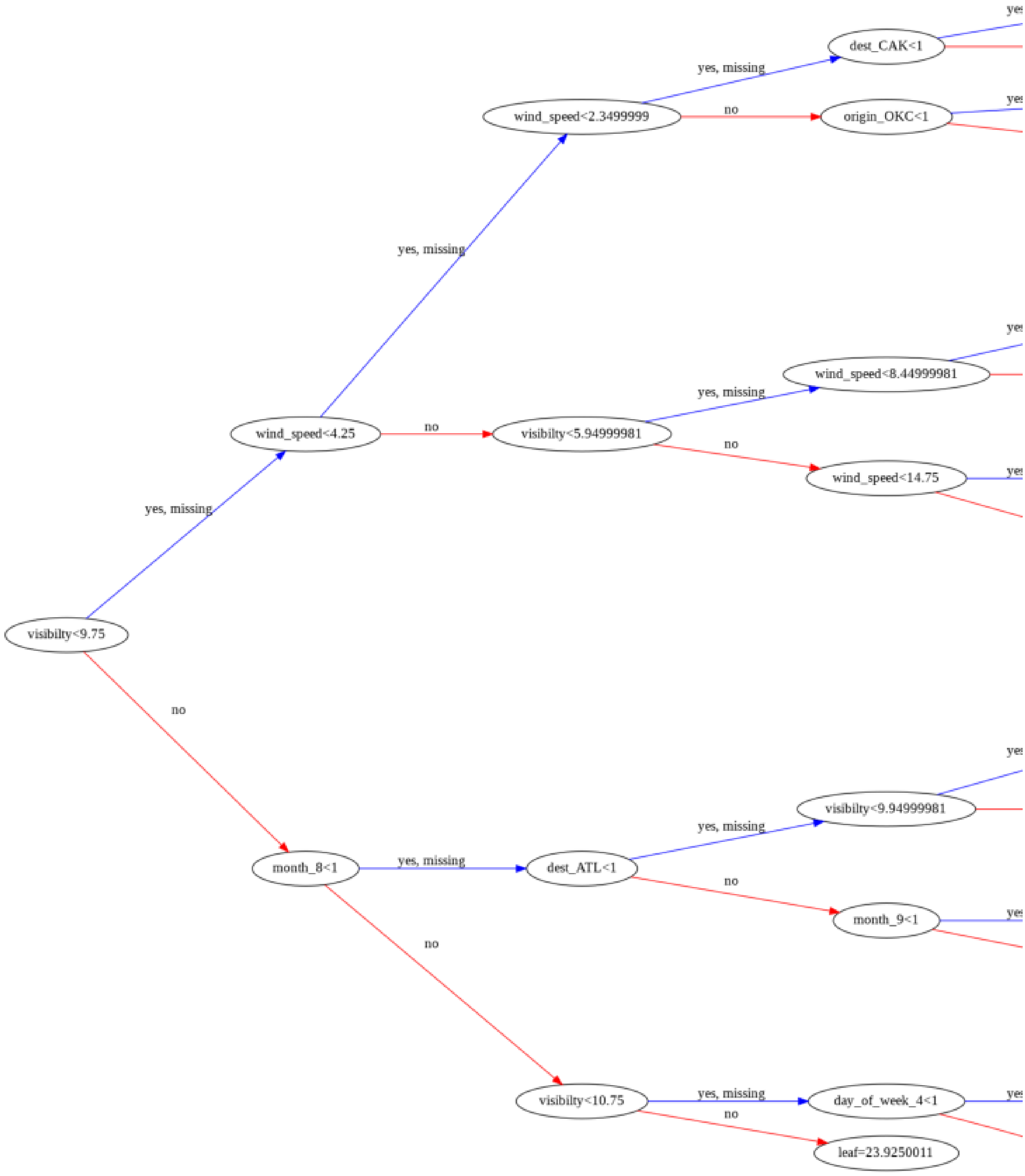
```
xg_reg = xgb.train(params=params, dtrain=data_dmatrix, num_boost_round=10)
```

```
[18:04:26] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor
```

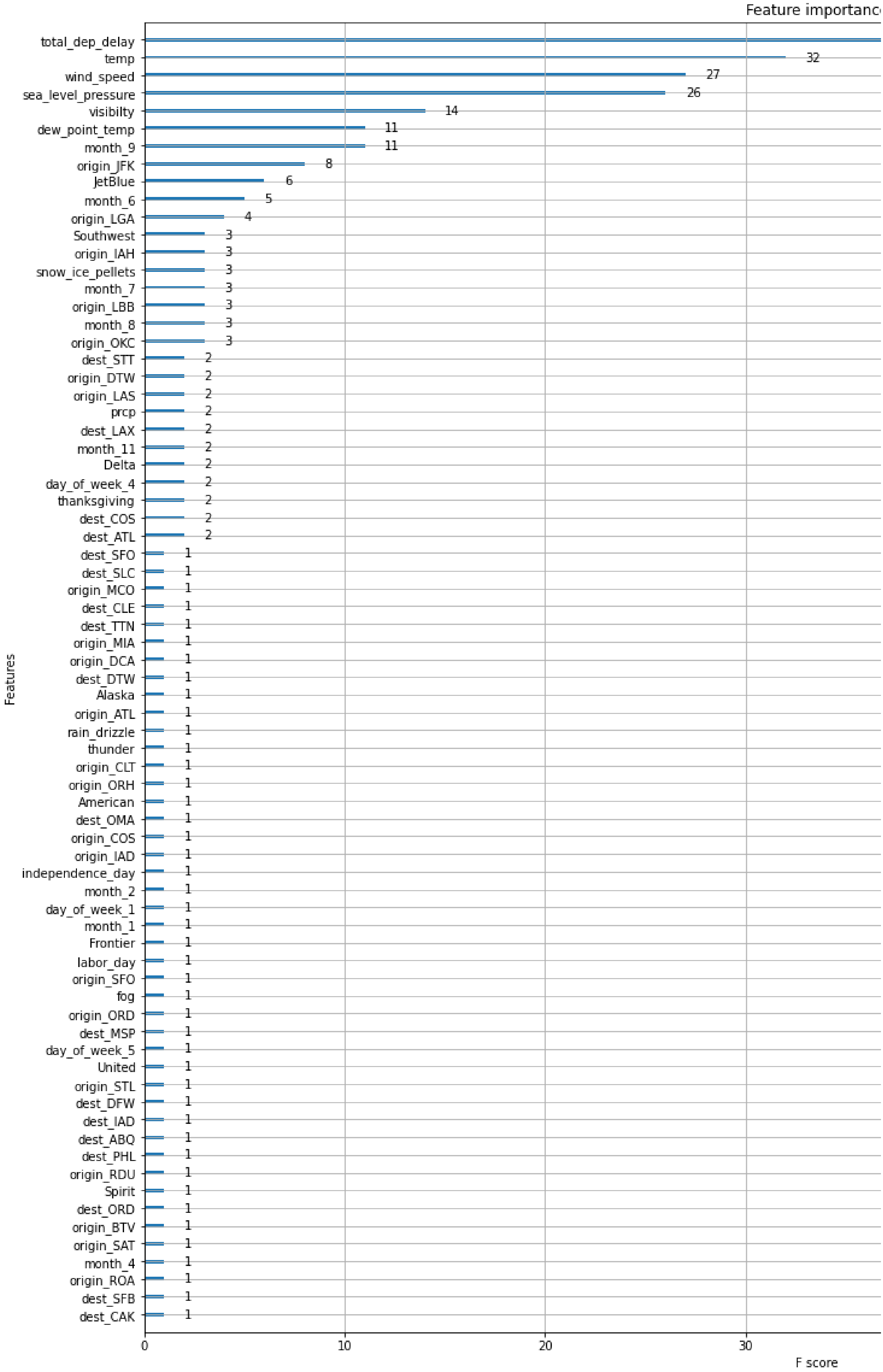
```
from numpy import loadtxt
from xgboost import XGBClassifier
from xgboost import plot_tree
import matplotlib.pyplot as plt

plt.rcParams["figure.figsize"] = 20,20
plot_tree(xg_reg, num_trees=0, rankdir='LR')
# xgb.plot_tree(xg_reg,num_trees=0)
# xgb.plot_tree(xg_reg, num_trees=0, rankdir='LR')
# plt.show()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f0c30acaeb8>
```



```
xgb.plot_importance(xg_reg)
plt.rcParams["figure.figsize"] = 20,20
plt.show()
```


Feature importanc
Features
total_dep_delay
temp 32
wind_speed 27
sea_level_pressure 26
visibilty 14
dew_point_temp 11
month_9 11
origin_JFK 8
JetBlue 6
month_6 5
origin_LGA 4
Southwest 3
origin_IAH 3
snow_ice_pellets 3
month_7 3
origin_LBB 3
month_8 3
origin_OKC 3
dest_STT 2
origin_DTW 2
origin_LAS 2
prcp 2
dest_LAX 2
month_11 2
Delta 2
day_of_week_4 2
thanksgiving 2
dest_COS 2
dest_ATL 2
dest_SFO 1
dest_SLC 1
origin_MCO 1
dest_CLE 1
dest_TTN 1
origin_MIA 1
origin_DCA 1
dest_DTW 1
Alaska 1
origin_ATL 1
rain_drizzle 1
thunder 1
origin_CLT 1
origin_ORH 1
American 1
dest_OMA 1
origin_COS 1
origin_IAD 1
independence_day 1
month_2 1
day_of_week_1 1
month_1 1
Frontier 1
labor_day 1
origin_SFO 1
fog 1
origin_ORD 1
dest_MSP 1
day_of_week_5 1
United 1
origin_STL 1
dest_DFW 1
dest_IAD 1
dest_ABQ 1
dest_PHL 1
origin_RDU 1
Spirit 1
dest_ORD 1
origin_BTV 1
origin_SAT 1
month_4 1
origin_ROA 1
dest_SFB 1
dest_CAK 1
0
10
20
30
F score

```
import pandas as pd
data = pd.read_csv(’big_data.csv’)
pd.DataFrame.from_records(data)
```

```
import pandas as pd
from sklearn.model_selection import train_test_split
import xgboost as xgb
from sklearn.metrics import mean_squared_error
import numpy as np

data.dropna(axis=0, subset=[’total_dep_delay’], inplace=True)

y = data.total_dep_delay
X = data.drop([’total_dep_delay’], axis=1).select_dtypes(exclude=[’object’])


X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=123)


data_dmatrix = xgb.DMatrix(data=X,label=y)


xg_reg = xgb.XGBRegressor(objective =’reg:linear’, colsample_bytree = 0.3, learning_rate = 0.1,
                          max_depth = 5, alpha = 10, n_estimators = 10)


xg_reg.fit(X_train,y_train)

preds = xg_reg.predict(X_test)
```

```
[17:50:08] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor
```

```
rmse = np.sqrt(mean_squared_error(y_test, preds))
print(″RMSE: %f″ % (rmse))
```

```
RMSE: 32.142434
```

changes for model tuning

```
params = {″objective″:″reg:linear″,’colsample_bytree’: 0.3,’learning_rate’: 0.1,
                ’max_depth’: 5, ’alpha’: 10}

cv_results = xgb.cv(dtrain=data_dmatrix, params=params, nfold=3,
                    num_boost_round=50,early_stopping_rounds=10,metrics=″rmse″, as_pandas=
```

```
[17:51:26] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor
[17:51:29] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor
[17:51:33] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor
```

```
cv_results.head()
```

	train-rmse-mean	train-rmse-std	test-rmse-mean	test-rmse-std
0	38.656838	0.041707	38.663035	0.085181
3	35.358801	0.054765	35.367652	0.076687

```
print((cv_results["test-rmse-mean"]).tail(1))
```

```
49    11.025499
Name: test-rmse-mean, dtype: float64
```

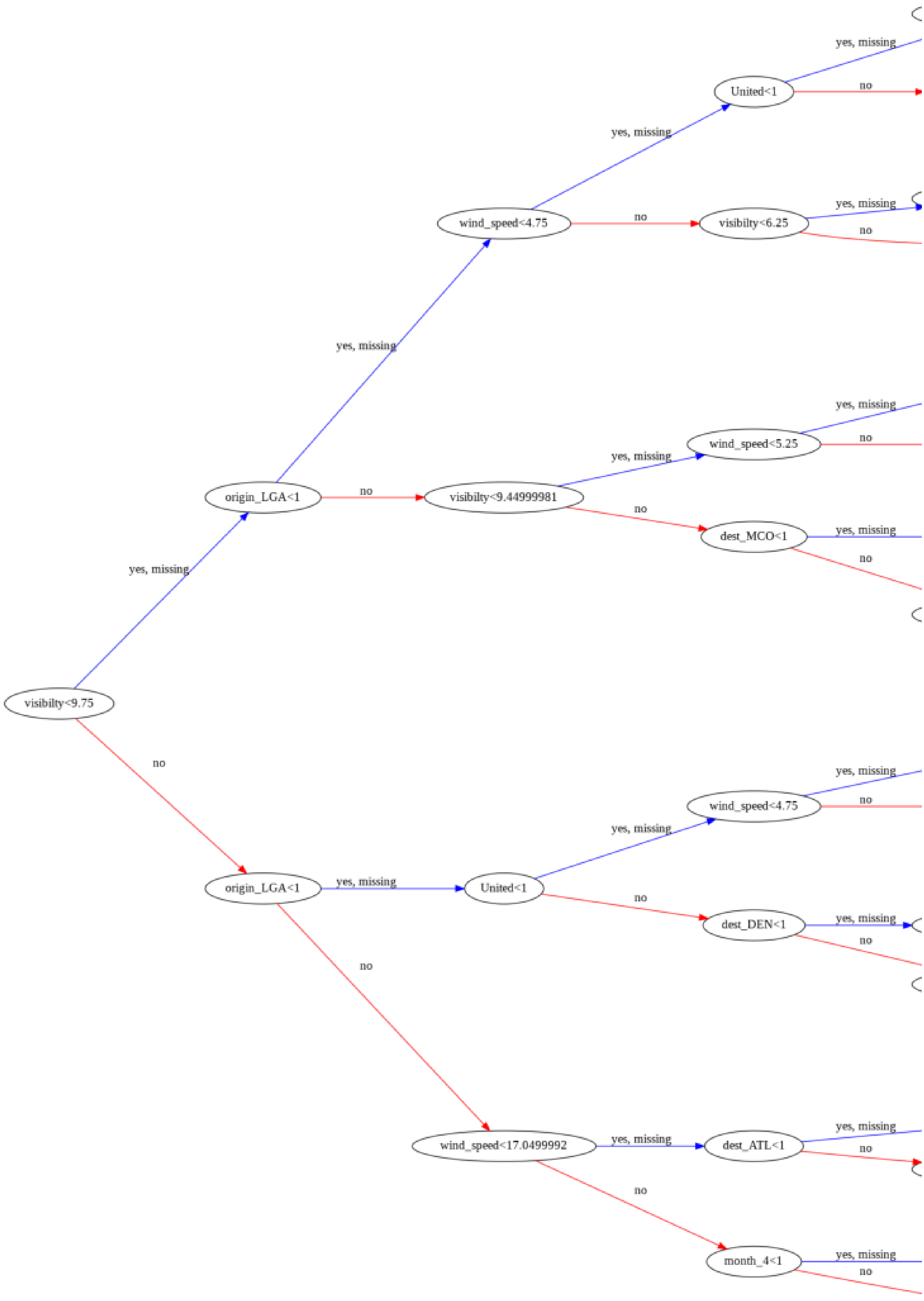
```
xg_reg = xgb.train(params=params, dtrain=data_dmatrix, num_boost_round=10)
```

```
[17:53:26] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in favor
```

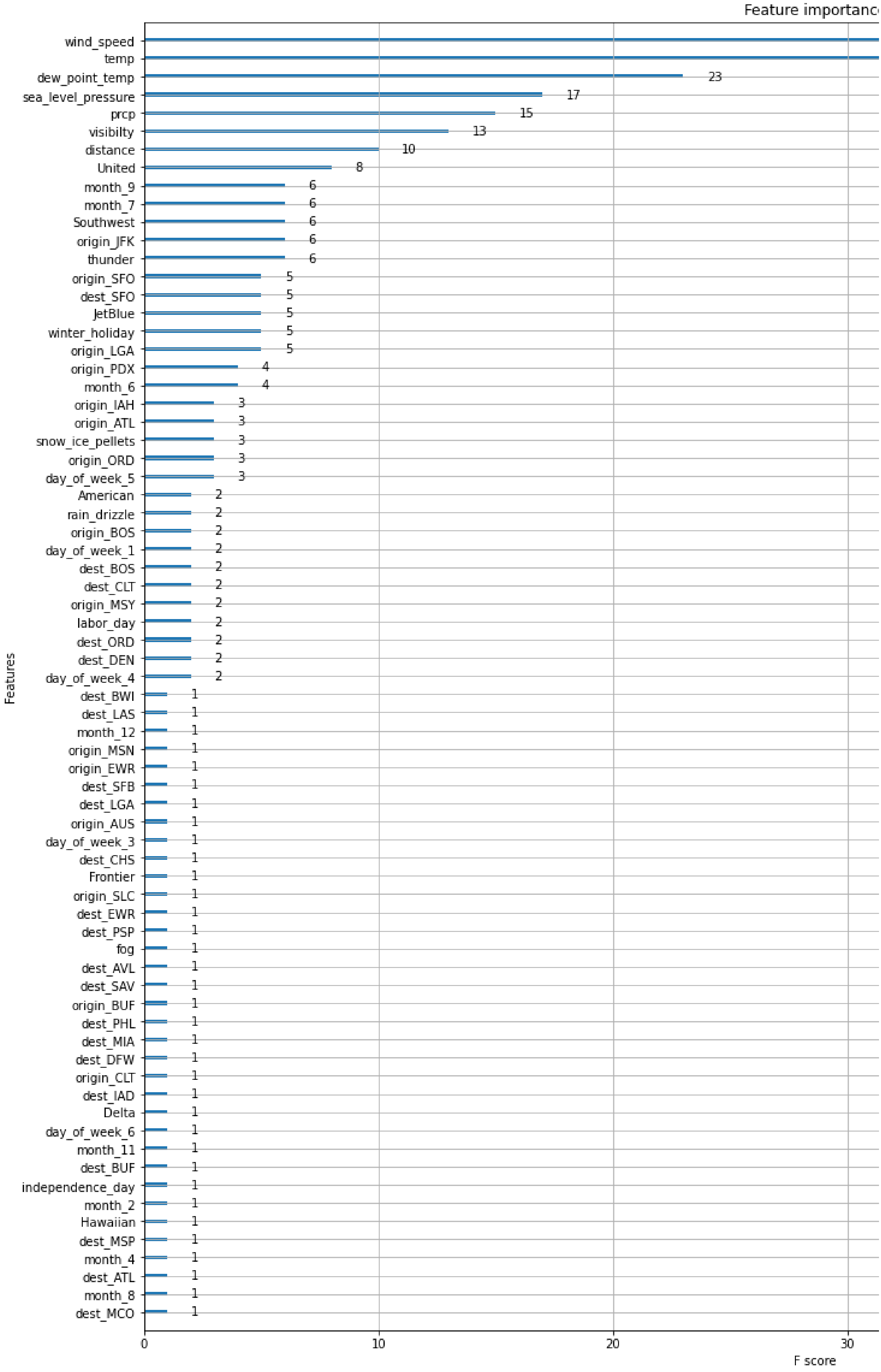
```
from numpy import loadtxt
from xgboost import XGBClassifier
from xgboost import plot_tree
import matplotlib.pyplot as plt

plt.rcParams["figure.figsize"] = 20,20
plot_tree(xg_reg, num_trees=0, rankdir='LR')
# xgb.plot_tree(xg_reg,num_trees=0)
# xgb.plot_tree(xg_reg, num_trees=0, rankdir='LR')
# plt.show()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f0c30a44978>
```


visibilty<9.75
yes, missing
no
origin_LGA<1
origin_LGA<1
wind_speed<4.75
visibilty<9.44999981
United<1
wind_speed<17.0499992
United<1
visibilty<6.25
wind_speed<5.25
dest_MCO<1
wind_speed<4.75
dest_DEN<1
dest_ATL<1
month_4<1

```
xgb.plot_importance(xg_reg)
plt.rcParams[’figure.figsize’] = [5, 5]
plt.show()
```


Feature importanc
wind_speed
temp
dew_point_temp
23
sea_level_pressure
17
prcp
15
visibilty
13
distance
10
United
8
month_9
6
month_7
6
Southwest
6
origin_JFK
6
thunder
6
origin_SFO
5
dest_SFO
5
JetBlue
5
winter_holiday
5
origin_LGA
5
origin_PDX
4
month_6
4
origin_IAH
3
origin_ATL
3
snow_ice_pellets
3
origin_ORD
3
day_of_week_5
3
American
2
rain_drizzle
2
origin_BOS
2
day_of_week_1
2
dest_BOS
2
dest_CLT
2
origin_MSY
2
labor_day
2
dest_ORD
2
dest_DEN
2
day_of_week_4
2
dest_BWI
1
dest_LAS
1
month_12
1
origin_MSN
1
origin_EWR
1
dest_SFB
1
dest_LGA
1
origin_AUS
1
day_of_week_3
1
dest_CHS
1
Frontier
1
origin_SLC
1
dest_EWR
1
dest_PSP
1
fog
1
dest_AVL
1
dest_SAV
1
origin_BUF
1
dest_PHL
1
dest_MIA
1
dest_DFW
1
origin_CLT
1
dest_IAD
1
Delta
1
day_of_week_6
1
month_11
1
dest_BUF
1
independence_day
1
month_2
1
Hawaiian
1
dest_MSP
1
month_4
1
dest_ATL
1
month_8
1
dest_MCO
1
Features
0
10
20
30
F score