Prediction of Fine Dust by Weather

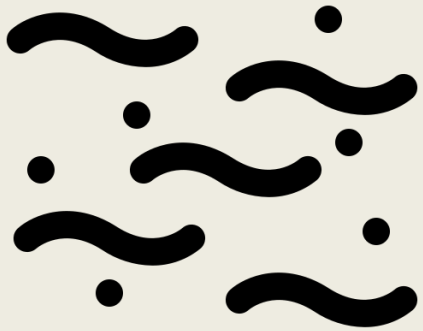
22000120 Kim Shin
22000350 Seo Jueun
22000532 Lee Seungjae
22000690 Jung Yiju

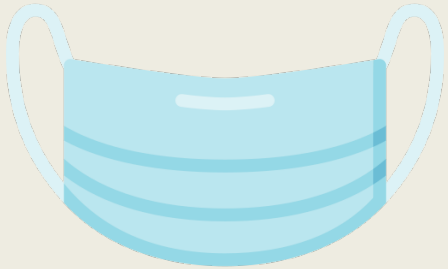
CONTENTS

Problem Definition

The average annual fine dust level is more than OECD average

Problem Definition



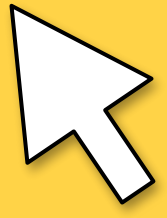


Fine dust causes
conjunctivitis, rhinitis, and bronchitis

Problem Definition

People want to predict
the concentration of fine dust
and prepare for it

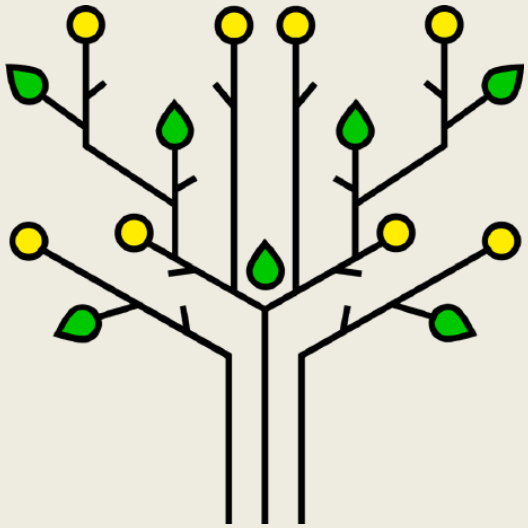
Background



Air quality information
and weather information

Linear prediction techniques,
nonlinear techniques using machine learning
and time series models

Proposed Method
(Random Forest)



- Form multiple decision trees
- New data points pass through each tree at the same time
- Each tree will vote on the results classified to select the result with the most votes as the final classification result

Proposed Method
(Random Forest)

	PM10	일시	평균풍속 (m/s)	최대풍속 (m/s)	최대풍속풍향 (deg)	평균기온 (℃)	평균최고기온 (℃)	평균최저기온 (℃)	강수량 (mm)	평균습도 (%rh)	최저습도 (%rh)	일[illegible]합 [illegible]r)	일[illegible]율 [illegible]%)	일[illegible]합 (MJ/[illegible]2)
0	60.0	Jan-12	2.5	7.5	320	-2.8	1.3	-6.3	6.7	49.0	12.0	1[illegible]0.5	6[illegible]05	23[illegible]50
1	50.0	Feb-12	2.9	8.1	290	-2.0	3.0	-6.0	0.8	43.0	11.0	2[illegible]4.9	7[illegible]74	32[illegible]25
2	47.0	Mar-12	3.5	10.2	290	5.1	9.5	1.5	47.4	52.0	9.0	1[illegible]1.8	5[illegible]63	36[illegible]51
3	51.0	Apr-12	3.4	12.0	290	12.3	17.9	7.8	157.0	54.0	9.0	2[illegible]2.5	5[illegible]72	44[illegible]48
4	52.0	May-12	2.7	8.6	250	19.7	25.1	15.4	8.2	48.0	11.0	2[illegible]0.3	5[illegible]04	55[illegible]40
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
115	21.0	Aug-21	2.1	8.3	320	25.9	29.7	22.8	211.2	74.0	39.0	1[illegible]7.3	3[illegible]24	42[illegible]88
116	15.0	Sep-21	2.3	7.1	320	22.6	26.9	18.8	131.0	71.0	38.0	1[illegible]2.0	4[illegible]82	47[illegible]13
117	27.0	Oct-21	2.1	10.6	290	15.6	20.5	11.6	57.0	70.0	25.0	1[illegible]3.4	4[illegible]38	35[illegible]62
118	46.0	Nov-21	2.1	9.3	270	8.2	13.1	4.0	62.4	68.0	27.0	1[illegible]3.6	5[illegible]57	27[illegible]64
119	39.0	Dec-21	2.3	8.1	290	0.6	5.1	-3.5	7.9	62.0	26.0	1[illegible]5.4	6[illegible]07	25[illegible]25

120 rows × 14 columns

In [158]:
```
#기본적인 randomforest모형

from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score # 정확도 함수

clf = RandomForestClassifier(n_estimators=100, max_depth=100,random_state=0)
clf.fit(train_x,train_y)

predict1 = clf.predict(test_x)
print(accuracy_score(test_y,predict1))
```

```
0.3333333333333333
```

Proposed Method
(Multiple Linear Regression)

Linear approach for modeling the relationship
between a scalar response
and one or more explanatory variables

Add solar radiation data

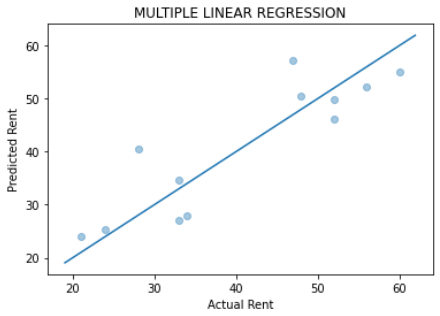
[348]:
```
x = df[['평균풍속(m/s)', '최대풍속(m/s)', '최대풍속풍향(deg)', '평균기온(℃)', '평균최고기온(℃)', '평균최저기온(℃)',
        '강수량(mm)', '평균습도(%rh)', '최저습도(%rh)', '일조합(hr)', '일조율(%)', '일사합(MJ/m2)']]
y = df[['PM10']]
x_train, x_test, y_train, y_test = train_test_split(x, y, train_size=0.9, test_size=0.1)
```

[349]:
```
mlr = LinearRegression()
mlr.fit(x_train, y_train)
```

[349]:
```
LinearRegression()
```

[350]:
```
y_predict = mlr.predict(x_test)
```

[351]:
```
plt.scatter(y_test, y_predict, alpha=0.4)
x = y = plt.xlim()
plt.plot(x, y)
plt.xlabel("Actual Rent")
plt.ylabel("Predicted Rent")
plt.title("MULTIPLE LINEAR REGRESSION")
plt.show()
```



[352]:
```
accuracy = mlr.score(x_test, y_test)
print("{:.2f}%".format(accuracy * 100))
```

```
77.86%
```

Exclude solar radiation data

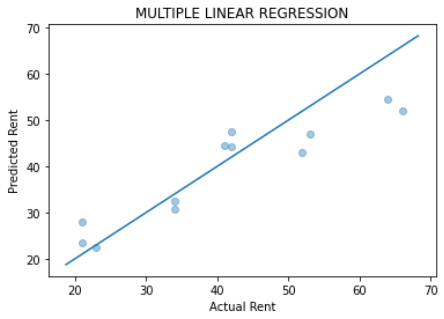
[468]:
```
x = df[['평균풍속(m/s)', '최대풍속(m/s)', '최대풍속풍향(deg)', '평균기온(℃)', '평균최고기온(℃)', '평균최저기온(℃)',
        '강수량(mm)', '평균습도(%rh)', '최저습도(%rh)']]
y = df[['PM10']]
x_train, x_test, y_train, y_test = train_test_split(x, y, train_size=0.9, test_size=0.1)
```

[469]:
```
mlr = LinearRegression()
mlr.fit(x_train, y_train)
```

[469]:
```
LinearRegression()
```

[470]:
```
y_predict = mlr.predict(x_test)
```

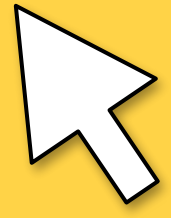
[471]:
```
plt.scatter(y_test, y_predict, alpha=0.4)
x = y = plt.xlim()
plt.plot(x, y)
plt.xlabel("Actual Rent")
plt.ylabel("Predicted Rent")
plt.title("MULTIPLE LINEAR REGRESSION")
plt.show()
```



[472]:
```
accuracy = mlr.score(x_test, y_test)
print("{:.2f}%".format(accuracy * 100))
```

```
80.51%
```

Result



Exclude the data related to the solar radiation with X variables

Linear regression techniques are more effective

Accuracy about 70 to 80 percent


-
x
Thank You