

## 808: Harnessing Data Analytics for Pollution level forecasting using the air quality attributes.

First Name	Last Name	Email (hawk.iit.edu)	Student ID
Srujan Shekar	Shetty	<a href="mailto:Sshetty16@hawk.iit.edu">Sshetty16@hawk.iit.edu</a>	<b>A20529733</b>
Senthilvel	Rajakarthihan		A20555853

### Table of Contents

<b>1. Introduction</b> .....	2
<b>2. Data</b> .....	3
<b>3. Problems to be Solved</b> .....	4
<b>4. Solutions</b> .....	5
<b>5. Experiments and Results</b> .....	6
5.1. Methods and Process .....	6
5.2. Evaluations and Results .....	16
5.3. Findings .....	20
<b>6. Conclusions and Future Work</b> .....	22
6.1. Conclusions .....	23
6.2. Limitations .....	23
6.3. Potential Improvements or Future Work .....	23

## 1. Introduction

Introduce the background and motivations

**Background & Motivation** :Concern over air quality and its effects on public health is growing along with urbanization. It has been determined that one of the main causes of air pollution and its detrimental impacts on health is the existence of fine particulate matter, specifically PM2.5. With an emphasis on the Beijing PM2.5 dataset, the goal of this project is to create a pollution level forecasting model using air quality parameters. The goal of this project is to give precise and timely PM2.5 concentration predictions so that preventative actions can be taken to lessen the health risks related to poor air quality. Our goal is to promote public health and contribute to environmental monitoring by utilizing data analysis techniques.

## 2. Data

Introduce your data, such as where did you get it (provide the URL if possible), how large it is, what are the variables/features, what are the variable types, etc

The data is from datasciencedojo-

<https://code.datasciencedojo.com/datasciencedojo/datasets/tree/master/Beijing%20PM2.5>

The size of the data -Row Size- 43825; Column Size-13.

### Data Dictionary

Column Position	Attribute Name	Definition	Data Type
1	No	No: row number	Quantitative
2	Year	Year: year of data in this row	Quantitative
3	Month	Month: month of data in this row	Quantitative
4	Day	Day: day of data in this row	Quantitative
5	Hour	Hour: hour of data in this row	Quantitative
6	PM2.5	PM2.5: PM2.5 concentration (ug/m <sup>3</sup> )	Quantitative
7	DEWP	DEWP: Dew Point (°f)	Quantitative
8	TEMP	TEMP: Temperature (°f)	Quantitative
9	PRES	PRES: Pressure (hPa)	Quantitative
10	cbwd	cbwd: Combined wind direction	Quantitative
11	lws	lws: Cumulated wind speed (m/s)	Quantitative
12	lr	lr: Cumulated hours of snow	Quantitative
13	ls	lr: Cumulated hours of rain	Quantitative

### 3. Problems to be Solved

*List the problems you want to solve*

To estimate the PM2.5 which is air quality index(dependent variable) of the Beijing city based on independent variables such as DEWP: Dew Point ( $^{\circ}\text{f}$ ),TEMP: Temperature ( $^{\circ}\text{f}$ ), PRES: Pressure (hPa),cbwd: Combined wind direction,lws: Cumulated wind speed (m/s),ls: Cumulated hours of snow,lr: Cumulated hours of rain,Hour,Day,Year,Month.

Dataset also deals with following problems:

- Missing values in Dataset
- Duplicate rows in Dataset

## 4. Solutions

You can use linear regression to predict air quality(Pm2.5),  
Independent variables such as DEWP: Dew Point ( $\hat{a},f$ ),TEMP: Temperature ( $\hat{a},f$ ),  
PRES: Pressure (hPa),cbwd: Combined wind direction,lws: Cumulated wind speed (m/s),ls:  
Cumulated hours of snow,lr: Cumulated hours of rain,Hour,Day,Year,Month.  
and air quality acting as the dependent variable(Pm2.5),.

The get to the solution ,problem is split into 3 parts:

### **1.Data Pre-Processing**

- 1.1 Exploring Data-Set
- 1.2 Filling Missing Values in Dataset with mean
- 1.3 Duplicate Rows Check
- 1.4 Correlation of Dataset
- 1.5 Using hold-out evaluation only, 80% as training

### **2.Linear Regression**

- 2.1 Full model
- 2.2 Backward method using p-value in t-test as metric.
- 2.3 Backward method using AIC as metric
- 2.4 Forward method using AIC as metric
- 2.5 Stepwise method using AIC as a metric

### **3.Post Processing**

- 3.1 Best Model from Linear Regression(Skipped RMSE Output Slide in PPT)
- 3.2 Model Diagnosis
- 3.3 Improving Model(Multi-Collinearity output slide,5 Cross Validation)

The best fit linear regression model can be used to predict air quality, with Dew Point ( $\hat{a},f$ ),TEMP: Temperature ( $\hat{a},f$ ), and other property qualities acting as independent factors and air quality acting as the dependent variable.

## 5. Experiments and Results

### 5.1. Methods and Process

#### 1.Data Pre-Processing

##### 1.1 Exploring Data-Set

To understand the dataset better we explored the data

The Code(comments explain the code):

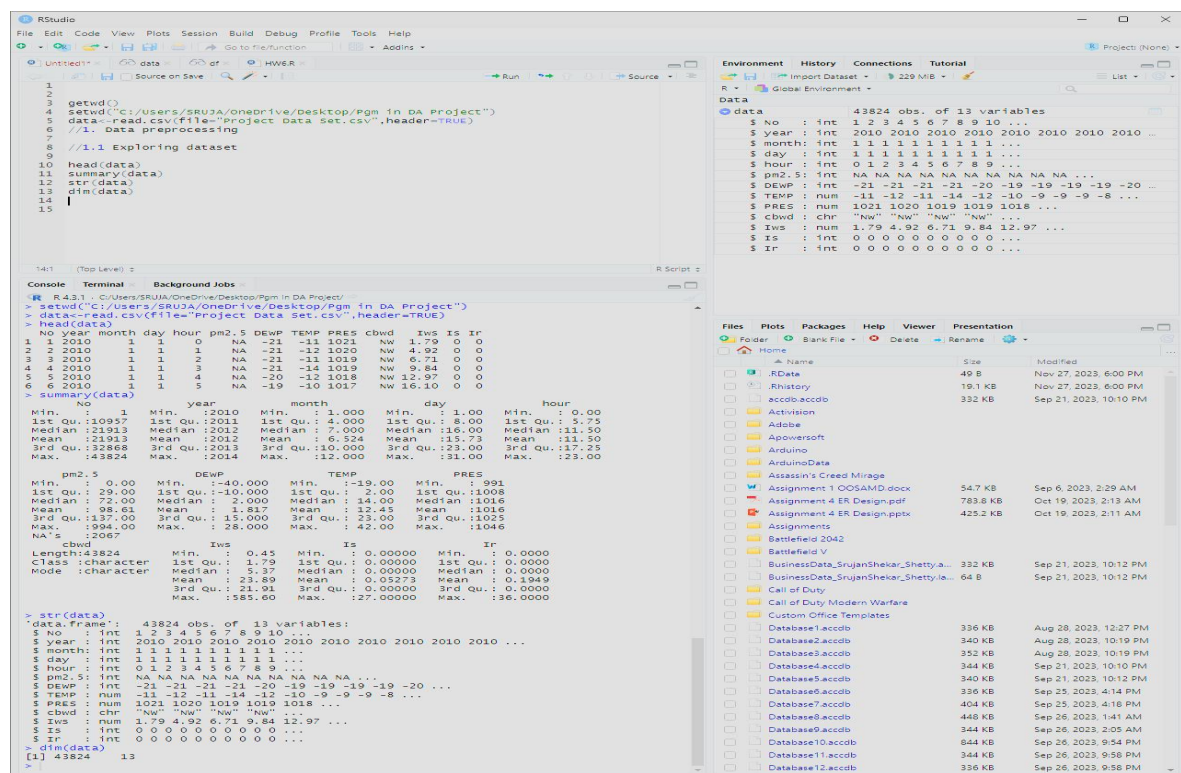
`head(data)` # displays the top few rows of your dataset.

`summary(data)` # summary of the dataset's variables' respective statistical data.

`str(data)` #displays data types giving the dataset's structure

`dim(data)` #displays dataset's number of rows and columns

The Snapshot:



```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418  
419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485  
486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539  
540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593  
594  
595  
596  
597  
598  
599  
600  
601  
602  
603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658  
659  
660  
661  
662  
663  
664  
665  
666  
667  
668  
669  
670  
671  
672  
673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689  
690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755  
756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768  
769  
770  
771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782  
783  
784  
785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796  
797  
798  
799  
800  
801  
802  
803  
804  
805  
806  
807  
808  
809  
810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863  
864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
918  
919  
920  
921  
922  
923  
924  
925  
926  
927  
928  
929  
930  
931  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971  
972  
973  
974  
975  
976  
977  
978  
979  
980  
981  
982  
983  
984  
985  
986  
987  
988  
989  
990  
991  
992  
993  
994  
995  
996  
997  
998  
999  
1000  
1001  
1002  
1003  
1004  
1005  
1006  
1007  
1008  
1009  
1010  
1011  
1012  
1013  
1014  
1015  
1016  
1017  
1018  
1019  
1020  
1021  
1022  
1023  
1024  
1025  
1026  
1027  
1028  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1050  
1051  
1052  
1053  
1054  
1055  
1056  
1057  
1058  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079  
1080  
1081  
1082  
1083  
1084  
1085  
1086  
1087  
1088  
1089  
1090  
1091  
1092  
1093  
1094  
1095  
1096  
1097  
1098  
1099  
1100  
1101  
1102  
1103  
1104  
1105  
1106  
1107  
1108  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1120  
1121  
1122  
1123  
1124  
1125  
1126  
1127  
1128  
1129  
1130  
1131  
1132  
1133  
1134  
1135  
1136  
1137  
1138  
1139  
1140  
1141  
1142  
1143  
1144  
1145  
1146  
1147  
1148  
1149  
1150  
1151  
1152  
1153  
1154  
1155  
1156  
1157  
1158  
1159  
1160  
1161  
1162  
1163  
1164  
1165  
1166  
1167  
1168  
1169  
1170  
1171  
1172  
1173  
1174  
1175  
1176  
1177  
1178  
1179  
1180  
1181  
1182  
1183  
1184  
1185  
1186  
1187  
1188  
1189  
1190  
1191  
1192  
1193  
1194  
1195  
1196  
1197  
1198  
1199  
1200  
1201  
1202  
1203  
1204  
1205  
1206  
1207  
1208  
1209  
1210  
1211  
1212  
1213  
1214  
1215  
1216  
1217  
1218  
1219  
1220  
1221  
1222  
1223  
1224  
1225  
1226  
1227  
1228  
1229  
1230  
1231  
1232  
1233  
1234  
1235  
1236  
1237  
1238  
1239  
1240  
1241  
1242  
1243  
1244  
1245  
1246  
1247  
1248  
1249  
1250  
1251  
1252  
1253  
1254  
1255  
1256  
1257  
1258  
1259  
1260  
1261  
1262  
1263  
1264  
1265  
1266  
1267  
1268  
1269  
1270  
1271  
1272  
1273  
1274  
1275  
1276  
1277  
1278  
1279  
1280  
1281  
1282  
1283  
1284  
1285  
1286  
1287  
1288  
1289  
1290  
1291  
1292  
1293  
1294  
1295  
1296  
1297  
1298  
1299  
1300  
1301  
1302  
1303  
1304  
1305  
1306  
1307  
1308  
1309  
1310  
1311  
1312  
1313  
1314  
1315  
1316  
1317  
1318  
1319  
1320  
1321  
1322  
1323  
1324  
1325  
1326  
1327  
1328  
1329  
1330  
1331  
1332  
1333  
1334  
1335  
1336  
1337  
1338  
1339  
1340  
1341  
1342  
1343  
1344  
1345  
1346  
1347  
1348  
1349  
1350  
1351  
1352  
1353  
1354  
1355  
1356  
1357  
1358  
1359  
1360  
1361  
1362  
1363  
1364  
1365  
1366  
1367  
1368  
1369  
1370  
1371  
1372  
1373  
1374  
1375  
1376  
1377  
1378  
1379  
1380  
1381  
1382  
1383  
1384  
1385  
1386  
1387  
1388  
1389  
1390  
1391  
1392  
1393  
1394  
1395  
1396  
1397  
1398  
1399  
1400  
1401  
1402  
1403  
1404  
1405  
1406  
1407  
1408  
1409  
1410  
1411  
1412  
1413  
1414  
1415  
1416  
1417  
1418  
1419  
1420  
1421  
1422  
1423  
1424  
1425  
1426  
1427  
1428  
1429  
1430  
1431  
1432  
1433  
1434  
1435  
1436  
1437  
1438  
1439  
1440  
1441  
1442  
1443  
1444  
1445  
1446  
1447  
1448  
1449  
1450  
1451  
1452  
1453  
1454  
1455  
1456  
1457  
1458  
1459  
1460  
1461  
1462  
1463  
1464  
1465  
1466  
1467  
1468  
1469  
1470  
1471  
1472  
1473  
1474  
1475  
1476  
1477  
1478  
1479  
1480  
1481  
1482  
1483  
1484  
1485  
1486  
1487  
1488  
1489  
1490  
1491  
1492  
1493  
1494  
1495  
1496  
1497  
1498  
1499  
1500  
1501  
1502  
1503  
1504  
1505  
1506  
1507  
1508  
1509  
1510  
1511  
1512  
1513  
1514  
1515  
1516  
1517  
1518  
1519  
1520  
1521  
1522  
1523  
1524  
1525  
1526  
1527  
1528  
1529  
1530  
1531  
1532  
1533  
1534  
1535  
1536  
1537  
1538  
1539  
1540  
1541  
1542  
1543  
1544  
1545  
1546  
1547  
1548  
1549  
1550  
1551  
1552  
1553  
1554  
1555  
1556  
1557  
1558  
1559  
1560  
1561  
1562  
1563  
1564  
1565  
1566  
1567  
1568  
1569  
1570  
1571  
1572  
1573  
1574  
1575  
1576  
1577  
1578  
1579  
1580  
1581  
1582  
1583  
1584  
1585  
1586  
1587  
1588  
1589  
1590  
1591  
1592  
1593  
1594  
1595  
1596  
1597  
1598  
1599  
1600  
1601  
1602  
1603  
1604  
1605  
1606  
1607  
1608  
1609  
1610  
1611  
1612  
1613  
1614  
1615  
1616  
1617  
1618  
1619  
1620  
1621  
1622  
1623  
1624  
1625  
1626  
1627  
1628  
1629  
1630  
1631  
1632  
1633  
1634  
1635  
1636  
1637  
1638  
1639  
1640  
1641  
1642  
1643  
1644  
1645  
1646  
1647  
1648  
1649  
1650  
1651  
1652  
1653  
1654  
1655  
1656  
1657  
1658  
1659  
1660  
1661  
1662  
1663  
1664  
1665  
1666  
1667  
1668  
1669  
1670  
1671  
1672  
1673  
1674  
1675  
1676  
1677  
1678  
1679  
1680  
1681  
1682  
1683  
1684  
1685  
1686  
1687  
1688  
1689  
1690  
1691  
1692  
1693  
1694  
1695  
1696  
1697  
1698  
1699  
1700  
1701  
1702  
1703  
1704  
1705  
1706  
1707  
1708  
1709  
1710  
1711  
1712  
1713  
1714  
1715  
1716  
1717  
1718  
1719  
1720  
1721  
1722  
1723  
1724  
1725  
1726  
1727  
1728  
1729  
1730  
1731  
1732  
1733  
1734  
1735  
1736  
1737  
1738  
1739  
1740  
1741  
1742  
1743  
1744  
1745  
1746  
1747  
1748  
1749  
1750  
1751  
1752  
1753  
1754  
1755  
1756  
1757  
1758  
1759  
1760  
1761  
1762  
1763  
1764  
1765  
1766  
1767  
1768  
1769  
1770  
1771  
1772  
1773  
1774  
1775  
1776  
1777  
1778  
1779  
1780  
1781  
1782  
1783  
1784  
1785  
1786  
1787  
1788  
1789  
1790  
1791  
1792  
1793  
1794  
1795  
1796  
1797  
1798  
1799  
1800  
1801  
1802  
1803  
1804  
1805  
1806  
1807  
1808  
1809  
1810  
1811  
1812  
1813  
1814  
1815  
1816  
1817  
1818  
1819  
1820  
1821  
1822  
1823  
1824  
1825  
1826  
1827  
1828  
1829  
1830  
1831  
1832  
1833  
1834  
1835  
1836  
1837  
1838  
1839  
1840  
1841  
1842  
1843  
1844  
1845  
1846  
1847  
1848  
1849  
1850  
1851  
1852  
1853  
1854  
1855  
1856  
1857  
1858  
1859  
1860  
1861  
1862  
1863  
1864  
1865  
1866  
1867  
1868  
1869  
1870  
1871  
1872  
1873  
1874  
1875  
1876  
1877  
1878  
1879  
1880  
1881  
1882  
1883  
1884  
1885  
1886  
1887  
1888  
1889  
1890  
1891  
1892  
1893  
1894  
1895  
1896  
1897  
1898  
1899  
1900  
1901  
1902  
1903  
1904  
1905  
1906  
1907  
1908  
1909  
1910  
1911  
1912  
1913  
1914  
1915  
1916  
1917  
1918  
1919  
1920  
1921  
1922  
1923  
1924  
1925  
1926  
1927  
1928  
1929  
1930  
1931  
1932  
1933  
1934  
1935  
1936  
1937  
1938  
1939  
1940  
1941  
1942  
1943  
1944  
1945  
1946  
1947  
1948  
1949  
1950  
1951  
1952  
1953  
1954  
1955  
1956  
1957  
1958  
1959  
1960  
1961  
1962  
1963  
1964  
1965  
1966  
1967  
1968  
1969  
1970  
1971  
1972  
1973  
1974  
1975  
1976  
1977  
1978  
1979  
1980  
1981  
1982  
1983  
1984  
1985  
1986  
1987  
1988  
1989  
1990  
1991  
1992  
1993  
1994  
1995  
1996  
1997  
1998  
1999  
2000  
2001  
2002  
2003  
2004  
2005  
2006  
2007  
2008  
2009  
2010  
2011  
2012  
2013  
2014  
2015  
2016  
2017  
2018  
2019  
2020  
2021  
2022  
2023  
2024  
2025  
2026  
2027  
2028  
2029  
2030  
2031  
2032  
2033  
2034  
2035  
2036  
2037  
2038  
2039  
2040  
2041  
2042  
2043  
2044  
2045  
2046  
2047  
2048  
2049  
2050  
2051  
2052  
2053  
2054  
2055  
2056  
2057  
2058  
2059  
2060  
2061  
2062  
2063  
2064  
2065  
2066  
2067  
2068  
2069  
2070  
2071  
2072  
2073  
2074  
2075  
2076  
2077  
2078  
2079  
2080  
2081  
2082  
2083  
2084  
2085  
2086  
2087  
2088  
2089  
2090  
2091  
2092  
2093  
2094  
2095  
2096  
2097  
2098  
2099  
2100  
2101  
2102  
2103  
2104  
2105  
2106  
2107  
2108  
2109  
2110  
2111  
2112  
2113  
2114  
2115  
2116  
2117  
2118  
2119  
2120  
2121  
2122  
2123  
2124  
2125  
2126  
2127  
2128  
2129  
2130  
2131  
2132  
2133  
2134  
2135  
2136  
2137  
2138  
2139  
2140  
2141  
2142  
2143  
2144  
2145  
2146  
2147  
2148  
2149  
2150  
2151  
2152  
2153  
2154  
2155  
21
```

`colSums(is.na(data))` *#Total number of missing values in a column*

`data$pm2.5 <-ifelse(is.na(data$pm2.5),ave(data$pm2.5, FUN=`  
`function(x)mean(x,na.rm=TRUE)),data$pm2.5)` *#Fill missing values with mean values*

## The Snapshot:

The screenshot displays the RStudio interface with the following components:

- Source Editor:** Contains R code for data loading and preprocessing. The code includes comments and functions to handle missing values in the `pm2.5` column.
- Console:** Shows the execution of the code, including the output of `colSums(is.na(data))` and the successful execution of the `ifelse` function to fill missing values.
- Environment:** Displays the `data` object with 43824 observations and 13 variables. The variables and their data types are listed.
- Files:** Shows a file explorer view of the project directory, listing various files and folders.

**Source Editor Code:**

```
1  
2  
3 getwd()  
4 setwd("C:/Users/SRUJA/OneDrive/Desktop/Pgm in DA Project")  
5 data<-read.csv(file="Project Data Set.csv",header=TRUE)  
6 #1. data preprocessing  
7  
8 #1.1 Exploring dataset  
9  
10 head(data)  
11 summary(data)  
12 str(data)  
13 dim(data)  
14  
15 #1.2 Missing values  
16 colSums(is.na(data)) #Total number of missing values in a column  
17 data$pm2.5 <-ifelse(is.na(data$pm2.5),ave(data$pm2.5, FUN= function(x)mean(x,na.rm=TRUE)),  
18 data$pm2.5)  
19
```

**Console Output:**

```
R 4.3.1 - C:/Users/SRUJA/OneDrive/Desktop/Pgm in DA Project >  
> data<-read.csv(file="Project Data Set.csv",header=TRUE)  
> #1.1 Missing values  
> colSums(is.na(data)) #Total number of missing values in a column  
No year month day hour pm2.5 DEWP TEMP PRES cbwd Iws Is Ir  
0 0 0 0 0 2067 0 0 0 0 0 0 0  
> data$pm2.5 <-ifelse(is.na(data$pm2.5),ave(data$pm2.5, FUN= function(x)mean(x,na.rm=TRUE)),  
data$pm2.5)  
> #1.2 Missing values  
> colSums(is.na(data)) #Total number of missing values in a column  
No year month day hour pm2.5 DEWP TEMP PRES cbwd Iws Is Ir  
0 0 0 0 0 0 0 0 0 0 0 0 0  
> |
```

**Environment Data:**

Variable	Class	Values
\$ No	int	1 2 3 4 5 6 7 8 9 10 ...
\$ year	int	2010 2010 2010 2010 2010 2010 2010 2010 ...
\$ month	int	1 1 1 1 1 1 1 1 ...
\$ day	int	1 1 1 1 1 1 1 1 ...
\$ hour	int	0 1 2 3 4 5 6 7 8 9 ...
\$ pm2.5	num	98.6 98.6 98.6 98.6 98.6 ...
\$ DEWP	int	-21 -21 -21 -21 -20 -19 -19 -19 -20 ...
\$ TEMP	num	-11 -12 -11 -14 -12 -10 -9 -9 -8 ...
\$ PRES	num	1021 1020 1019 1019 1018 ...
\$ cbwd	chr	"Nw" "Nw" "Nw" "Nw" ...
\$ Iws	num	1.79 4.92 6.71 9.84 12.97 ...
\$ Is	int	0 0 0 0 0 0 0 0 ...
\$ Ir	int	0 0 0 0 0 0 0 0 ...

**Files:**

Name	Size	Modified
49 B	49 B	Nov 27, 2023, 6:00 PM
Rhistory	19.1 KB	Nov 27, 2023, 6:00 PM
acddb.acddb	332 KB	Sep 21, 2023, 10:10 PM
Activation		
Adobe		
Apowersoft		
Arduino		
Assassin's Creed Mirage		
Assignment 1 OOSAMD.docx	54.7 KB	Sep 6, 2023, 2:29 AM
Assignment 4 ER Design.pdf	783.8 KB	Oct 19, 2023, 2:13 AM
Assignment 4 ER Design.pptx	425.2 KB	Oct 19, 2023, 2:11 AM
Assignments		
Battlefield 2042		
Battlefield V		
BusinessData_SrujanShekar_Shetty.a...	332 KB	Sep 21, 2023, 10:12 PM
BusinessData_SrujanShekar_Shetty.la...	64 B	Sep 21, 2023, 10:12 PM
Call of Duty		
Call of Duty Modern Warfare		
Custom Office Templates		
Database1.acddb	336 KB	Aug 28, 2023, 12:27 PM
Database2.acddb	340 KB	Aug 28, 2023, 10:19 PM
Database3.acddb	352 KB	Aug 28, 2023, 10:19 PM
Database4.acddb	344 KB	Sep 21, 2023, 10:10 PM
Database5.acddb	340 KB	Sep 21, 2023, 10:12 PM
Database6.acddb	336 KB	Sep 25, 2023, 4:14 PM
Database7.acddb	404 KB	Sep 25, 2023, 4:18 PM
Database8.acddb	448 KB	Sep 26, 2023, 1:41 AM
Database9.acddb	344 KB	Sep 26, 2023, 2:05 AM
Database10.acddb	844 KB	Sep 26, 2023, 9:54 PM
Database11.acddb	344 KB	Sep 26, 2023, 9:58 PM
Database12.acddb	336 KB	Sep 26, 2023, 9:58 PM

### 1.3 Duplicate Rows Check

In order to avoid error Duplicate rows check was made

The Code (comments explain the code):

`any(duplicated(data))` *#check duplicacy*

The Snapshot:

The screenshot displays the RStudio interface with the following components:

- Source Editor:** Contains R code for data preprocessing:

```
1 # Read the data
2 data <- read.csv("C:/Users/Srujan/Desktop/fgm in DA Project/Project Data Set.csv", header=TRUE)
3 # Data preprocessing
4 #1.1 Exploring dataset
5 head(data)
6 summary(data)
7 dim(data)
8 #1.2 Missing values
9 columns <- na(data) #total number of missing values in a column
10 data$pm2.5 <- ifelse(is.na(data$pm2.5), ave(data$pm2.5, FUN= function(x) mean(x, na.rm=T)), data$pm2.5)
11 #1.3 Data duplicacy
12 any(duplicated(data))
```
- Console:** Shows the execution output:

```
R 4.3.1 : C:/Users/Srujan/Desktop/fgm in DA Project/
> data <- read.csv("C:/Users/Srujan/Desktop/fgm in DA Project/Project Data Set.csv", header=TRUE)
> #1.2 Missing values
> columns <- na(data) #total number of missing values in a column
no year month day hour pm2.5 DEWP TEMP PRES cbwd iwa ia ir
0 0 0 0 0 0 0 0 0 0 0 0
> data$pm2.5 <- ifelse(is.na(data$pm2.5), ave(data$pm2.5, FUN= function(x) mean(x, na.rm=TRUE)), data$pm2.5)
> #1.3 Missing values
> columns <- na(data) #total number of missing values in a column
no year month day hour pm2.5 DEWP TEMP PRES cbwd iwa ia ir
0 0 0 0 0 0 0 0 0 0 0 0
> any(duplicated(data))
[1] FALSE
```
- Environment:** Shows the 'data' object with 43824 observations and 13 variables.
- Files:** Lists various files in the project, including 'RData', 'acddb', 'Assassin's Creed Mirage', and several 'Database' files.



[illegible]

## 1.5 Using hold-out evaluation only, 80% as training

Estimating a machine learning model's performance on fresh, untested data is made easier with the aid of holdout evaluation.

The Code (comments explain the code):

```
data=data[sample(nrow(data)),]
```

```
select.data = sample (1:nrow(data), 0.8*nrow(data)) #80% as select data
```

```
train.data = data[select.data,] #Select data as train data
```

```
test.data = data[-select.data,] #Select data as
```

The screenshot displays the RStudio interface with the following components:

- Source Editor:** Contains R code for data sampling and model training. The code includes comments explaining the steps: excluding nominal variables, sampling 80% of the data for training, and fitting a linear model using backward selection with p-value and AIC metrics.
- Environment:** Lists the objects in the Global Environment: `data` (43824 obs. of 11 variables), `dummy_variables` (Large matrix), `m1` (Large lm), `test.data` (8765 obs. of 11 variables), and `train.data` (35059 obs. of 11 variables). The `values` section shows the `select.data` vector.
- Console:** Shows the execution of the R code, including the sampling process and the fitting of the linear model `m1`.
- Plots:** Displays a series of small plots, likely generated by the `summary(m1)` command, showing the distribution of variables and the model's performance.

**Hypothesis Testing for Individual Coefficients:**

Null Hypothesis ( $H_0$ ): There is no significant relationship between variables like DEWP, TEMP, etc., and PM2.5 concentration.

Alternative Hypothesis ( $H_1$ ): There is a significant relationship between variables like DEWP, TEMP, etc., and PM2.5 concentration.

Test Method: Perform z-tests for individual coefficients in a multiple linear regression model. The null hypothesis is rejected if the p-value is below a predetermined significance level (e.g., 0.05).

**Conclusion: We reject Null Hypothesis ( $H_0$ ) based on the test results**

## 2.Linear Regression

### 2.1 Full model

### 2.2 Backward method using p-value in t-test as metric.

### 2.3 Backward method using AIC as metric

### 2.4 Forward method using AIC as metric

### 2.5 Stepwise method using AIC as a metric

The Above Linear regression models are built in order to best fit the data in them and predict the unknown values of PM2.5.

The Best Model comes out to be Backward method using p-value in t-test as metric.

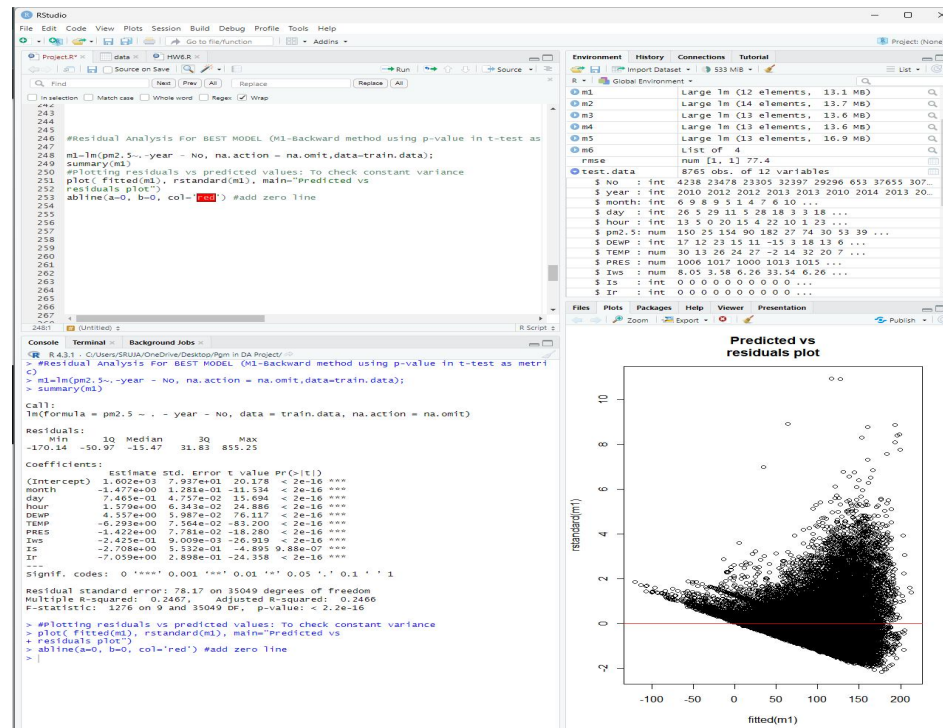
## 3.Post Processing

### 3.1 Best Model from Linear Regression

### 3.2 Model Diagnosis

Residual Analysis is considered for the following :

- Plot residuals vs predicted values: To Validate the constant variance

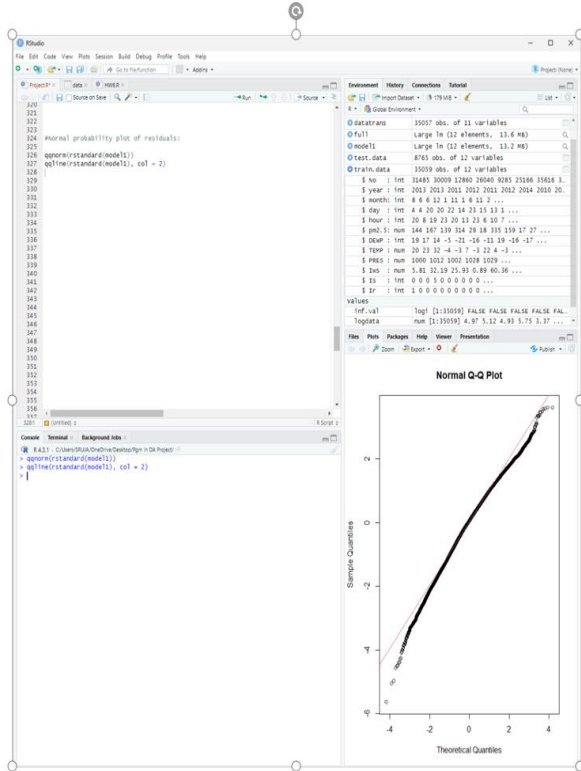


Transformation of PM2.5(Y variable):

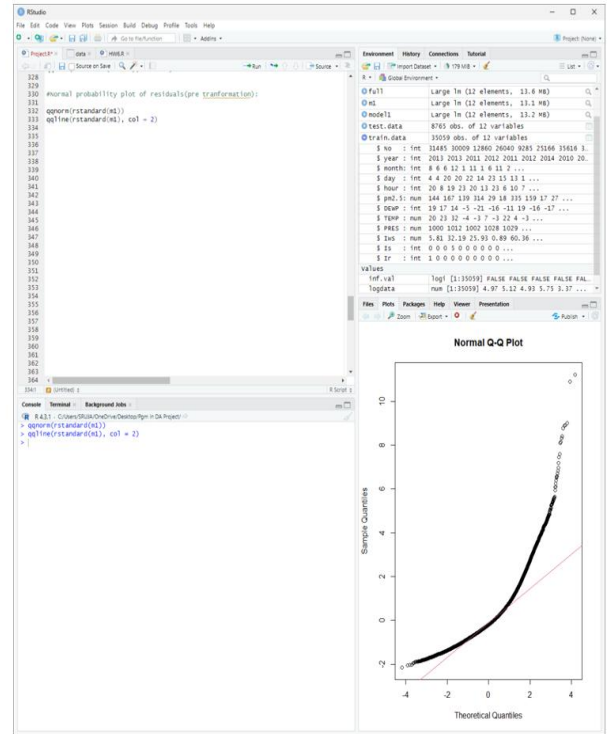
Transformation is done as the graph predicts a problem(pattern)

b) Plot residuals vs each x-variable: To Validate the linearity relationship

c) Normal probability plot of residuals: To check normality assumption for the error terms



QQ plot after Tranforming Pm2.5 variable.



QQ plot before Tranforming Pm2.5 variable.

### 3.3 Improving Model

#### a) Multicollinearity problem

#### Computing Variance Inflation Factor Statistics

The screenshot displays the RStudio interface with the following components:

- Source Editor:** Contains R code for evaluating collinearity and calculating VIFs.
- Environment:** Lists loaded objects including 'm1', 'model1', 'test.data', and 'train.data'.
- Console:** Shows the execution of the R script, including the installation of the 'car' package and the output of the VIF calculation.

```
# Evaluate collinearity
install.packages("car")
library(car)
vif(model1) # variance inflation factors
```

**Console Output:**

```
R 4.3.1 - C:/Users/SRUJA/OneDrive/Desktop/Pgm In DA Project/
https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/SRUJA/AppData/Local/R/win-library/4.3'
(as 'lib' is unspecified)
warning in install.packages :
  package 'vif' is not available for this version of R

A version of this package for your version of R might be available elsewhere,
see the ideas at
https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#installing-packages
> # Evaluate collinearity
> install.packages("car")
WARNING: Rtools is required to build R packages but is not currently installed. Please
download and install the appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/SRUJA/AppData/Local/R/win-library/4.3'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.3/car_3.1-2.zip'
Content type 'application/zip' length 1706777 bytes (1.6 MB)
downloaded 1.6 MB

package 'car' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:/Users/SRUJA/AppData/Local/Temp/RtmpK09T6Z/downloaded_packages
> library(car)
Loading required package: carData
warning message:
package 'car' was built under R version 4.3.2
> vif(model1) # variance inflation factors
   year   month   day   hour  DEWP   TEMP   PRES   IWS    IS
1.015430 1.121005 1.002727 1.105432 4.314122 4.936000 3.669711 1.147956 1.022161
  Ir
1.030083
> |
```

**Environment Panel:**

Object	Class	Size
m1	lm	Large (12 elements, 13.1 MB)
model1	lm	Large (12 elements, 13.2 MB)
test.data	data.frame	8765 obs. of 12 variables
train.data	data.frame	35059 obs. of 12 variables

**Values:**

Variable	log1 [1:35059]	FALSE	FALSE	FALSE	FAL...
inf.val	num [1:35059]	4.97	5.12	4.93	5.75 3.37 ...
logdata	num [1:35059]	38382	13346	17567	43074 3840...
select.data	int [1:35059]				

Since  $VIF > 4$  for DEWP, TEMP, we are testing corr to find pair of independent variables

The Correlation between DEWP and TEMP is  
**0.8244411847**

```
RStudio  
File Edit Code View Plots Session Build Debug Profile Tools Help  
Project: data | Source in Data | Run | Stop | Source | Environment | History | Connections | Tutorial | Project Change...  
43824 obs. of 12 variables  
logdata: num 4.97 5.12 4.95 5.75 5.37 ...  
year: fct 2013 2013 2013 2013 2013 2012 2012 2012 2012 2012 ...  
month: fct 8 6 6 12 1 11 1 6 11 2 ...  
day: fct 4 4 20 20 20 24 23 13 13 ...  
hour: fct 20 8 19 23 20 13 23 6 10 7 ...  
DEWP: fct 19 17 14 -3 -21 -16 -11 19 -16 -1 ...  
TEMP: num 20 21 12 -4 -13 7 -3 23 4 -3 ...  
PRES: num 1000 1002 1002 1028 1029 ...  
Is: fct 1 0 0 0 0 0 0 0 0 0 ...  
IR: fct 1 0 0 0 0 0 0 0 0 0 ...  
Full: Large Int (12 elements, 11.6 MB)  
model: Large Int (12 elements, 11.1 MB)  
multicols: Large Int (12 elements, 12.7 MB)  
test.data: 8765 obs. of 12 variables  
train.data: 35559 obs. of 12 variables  
File Plots Packages Help View Presentation  
File Edit View Plots Packages Help View Presentation  
Console Terminal Background Info  
R 4.3.1: C:\Users\GJL\OneDrive\Desktop>R -Dk.Rproj  
hour -0.020243792 0.002783938 -0.003893747 -0.001190102 1.000000000  
DEWP -0.11072174 0.000872774 0.23498760 0.025810312 -0.000237785  
TEMP -0.02288442 -0.047628204 0.170044857 0.023484953 0.150388646  
PRES -0.11803839 -0.0448910285 -0.006287731 -0.003042144 -0.040354287  
Is -0.14422994 -0.0044452785 0.000218202 -0.00887775 0.058609746  
IR -0.03092461 -0.0203278548 -0.05852343 -0.03473722 -0.002181126  
IR -0.04892124 -0.020654874 0.035330840 -0.003028999 -0.003404613  
logdata TEMP PRES Is IR  
logdata 0.1107217438 0.02288442 -0.118038387 -0.144229936 0.030924614  
year 0.0001873774 0.047628204 -0.048931019 -0.048465179 -0.020377815  
month 0.234987599 0.170044857 -0.046387733 0.001618200 -0.058521243  
day 0.025810312 0.023484953 -0.000606214 -0.008877753 -0.034737272  
hour -0.020237781 0.15038865 -0.040354197 0.056609746 -0.002181126  
DEWP 1.000000000 0.000000000 -0.71871028 -0.29743304 -0.031814602  
TEMP 0.8244411847 1.000000000 -0.82689614 -0.15014300 -0.091710942  
PRES -0.77871028 -0.82689614 1.000000000 0.18644904 0.08718791  
Is -0.29743304 -0.15014304 0.18644904 1.000000000 0.025775799  
IR -0.031814602 -0.09171094 0.08718791 0.025775799 1.000000000  
IR 0.125045672 0.04888862 -0.078842452 -0.00643099 -0.009414444  
logdata -0.048921243  
year -0.020654874  
month -0.035258403  
day -0.003028999  
hour -0.003404613  
DEWP 0.125045672  
TEMP 0.04888862  
PRES -0.078842452  
Is -0.00643099  
IR -0.009414444  
IR 1.000000000  
= multicols= logdata... data = dataframes[,4:1]  
> summary(multicols)  
call:  
lm(formula = logdata ~ ., data = dataframes[, -4])
```

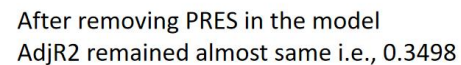
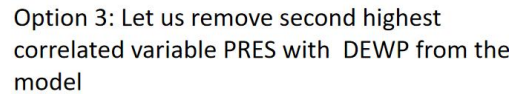
```
RStudio  
File Edit Code View Plots Session Build Debug Profile Tools Help  
Project: data | Source in Data | Run | Stop | Source | Environment | History | Connections | Tutorial | Project Change...  
43824 obs. of 12 variables  
logdata: num 4.97 5.12 4.95 5.75 5.37 ...  
year: fct 2013 2013 2013 2013 2013 2012 2012 2012 2012 2012 ...  
month: fct 8 6 6 12 1 11 1 6 11 2 ...  
day: fct 4 4 20 20 20 24 23 13 13 ...  
hour: fct 20 8 19 23 20 13 23 6 10 7 ...  
DEWP: fct 19 17 14 -3 -21 -16 -11 19 -16 -1 ...  
TEMP: num 20 21 12 -4 -13 7 -3 23 4 -3 ...  
PRES: num 1000 1002 1002 1028 1029 ...  
Is: fct 1 0 0 0 0 0 0 0 0 0 ...  
IR: fct 1 0 0 0 0 0 0 0 0 0 ...  
Full: Large Int (12 elements, 11.6 MB)  
model: Large Int (12 elements, 11.1 MB)  
multicols: Large Int (12 elements, 12.7 MB)  
test.data: 8765 obs. of 12 variables  
train.data: 35559 obs. of 12 variables  
File Plots Packages Help View Presentation  
File Edit View Plots Packages Help View Presentation  
Console Terminal Background Info  
R 4.3.1: C:\Users\GJL\OneDrive\Desktop>R -Dk.Rproj  
hour -0.020243792 0.002783938 -0.003893747 -0.001190102 1.000000000  
DEWP -0.11072174 0.000872774 0.23498760 0.025810312 -0.000237785  
TEMP -0.02288442 -0.047628204 0.170044857 0.023484953 0.150388646  
PRES -0.11803839 -0.0448910285 -0.006287731 -0.003042144 -0.040354287  
Is -0.14422994 -0.0044452785 0.000218202 -0.00887775 0.058609746  
IR -0.03092461 -0.0203278548 -0.05852343 -0.03473722 -0.002181126  
IR -0.04892124 -0.020654874 0.035330840 -0.003028999 -0.003404613  
logdata TEMP PRES Is IR  
logdata 0.1107217438 0.02288442 -0.118038387 -0.144229936 0.030924614  
year 0.0001873774 0.047628204 -0.048931019 -0.048465179 -0.020377815  
month 0.234987599 0.170044857 -0.046387733 0.001618200 -0.058521243  
day 0.025810312 0.023484953 -0.000606214 -0.008877753 -0.034737272  
hour -0.020237781 0.15038865 -0.040354197 0.056609746 -0.002181126  
DEWP 1.000000000 0.000000000 -0.71871028 -0.29743304 -0.031814602  
TEMP 0.8244411847 1.000000000 -0.82689614 -0.15014300 -0.091710942  
PRES -0.77871028 -0.82689614 1.000000000 0.18644904 0.08718791  
Is -0.29743304 -0.15014304 0.18644904 1.000000000 0.025775799  
IR -0.031814602 -0.09171094 0.08718791 0.025775799 1.000000000  
IR 0.125045672 0.04888862 -0.078842452 -0.00643099 -0.009414444  
logdata -0.048921243  
year -0.020654874  
month -0.035258403  
day -0.003028999  
hour -0.003404613  
DEWP 0.125045672  
TEMP 0.04888862  
PRES -0.078842452  
Is -0.00643099  
IR -0.009414444  
IR 1.000000000  
= multicols= logdata... data = dataframes[,4:1]  
> summary(multicols)  
call:  
lm(formula = logdata ~ ., data = dataframes[, -4])
```

Option 1: After removing DEWP in the model  
AdjR2 Dropped to 0.1797

```
RStudio  
File Edit Code View Plots Session Build Debug Profile Tools Help  
Project: data | Source in Data | Run | Stop | Source | Environment | History | Connections | Tutorial | Project Change...  
43824 obs. of 12 variables  
logdata: num 4.97 5.12 4.95 5.75 5.37 ...  
year: fct 2013 2013 2013 2013 2013 2012 2012 2012 2012 2012 ...  
month: fct 8 6 6 12 1 11 1 6 11 2 ...  
day: fct 4 4 20 20 20 24 23 13 13 ...  
hour: fct 20 8 19 23 20 13 23 6 10 7 ...  
DEWP: fct 19 17 14 -3 -21 -16 -11 19 -16 -1 ...  
TEMP: num 20 21 12 -4 -13 7 -3 23 4 -3 ...  
PRES: num 1000 1002 1002 1028 1029 ...  
Is: fct 1 0 0 0 0 0 0 0 0 0 ...  
IR: fct 1 0 0 0 0 0 0 0 0 0 ...  
Full: Large Int (12 elements, 11.6 MB)  
model: Large Int (12 elements, 11.1 MB)  
multicols: Large Int (12 elements, 12.7 MB)  
test.data: 8765 obs. of 12 variables  
train.data: 35559 obs. of 12 variables  
File Plots Packages Help View Presentation  
File Edit View Plots Packages Help View Presentation  
Console Terminal Background Info  
R 4.3.1: C:\Users\GJL\OneDrive\Desktop>R -Dk.Rproj  
hour -0.020243792 0.002783938 -0.003893747 -0.001190102 1.000000000  
DEWP -0.11072174 0.000872774 0.23498760 0.025810312 -0.000237785  
TEMP -0.02288442 -0.047628204 0.170044857 0.023484953 0.150388646  
PRES -0.11803839 -0.0448910285 -0.006287731 -0.003042144 -0.040354287  
Is -0.14422994 -0.0044452785 0.000218202 -0.00887775 0.058609746  
IR -0.03092461 -0.0203278548 -0.05852343 -0.03473722 -0.002181126  
IR -0.04892124 -0.020654874 0.035330840 -0.003028999 -0.003404613  
logdata TEMP PRES Is IR  
logdata 0.1107217438 0.02288442 -0.118038387 -0.144229936 0.030924614  
year 0.0001873774 0.047628204 -0.048931019 -0.048465179 -0.020377815  
month 0.234987599 0.170044857 -0.046387733 0.001618200 -0.058521243  
day 0.025810312 0.023484953 -0.000606214 -0.008877753 -0.034737272  
hour -0.020237781 0.15038865 -0.040354197 0.056609746 -0.002181126  
DEWP 1.000000000 0.000000000 -0.71871028 -0.29743304 -0.031814602  
TEMP 0.8244411847 1.000000000 -0.82689614 -0.15014300 -0.091710942  
PRES -0.77871028 -0.82689614 1.000000000 0.18644904 0.08718791  
Is -0.29743304 -0.15014304 0.18644904 1.000000000 0.025775799  
IR -0.031814602 -0.09171094 0.08718791 0.025775799 1.000000000  
IR 0.125045672 0.04888862 -0.078842452 -0.00643099 -0.009414444  
logdata -0.048921243  
year -0.020654874  
month -0.035258403  
day -0.003028999  
hour -0.003404613  
DEWP 0.125045672  
TEMP 0.04888862  
PRES -0.078842452  
Is -0.00643099  
IR -0.009414444  
IR 1.000000000  
= multicols= logdata... data = dataframes[,4:1]  
> summary(multicols)  
call:  
lm(formula = logdata ~ ., data = dataframes[, -4])
```

Option 2: After removing TEMP in the model  
AdjR2 Dropped to 0.1797



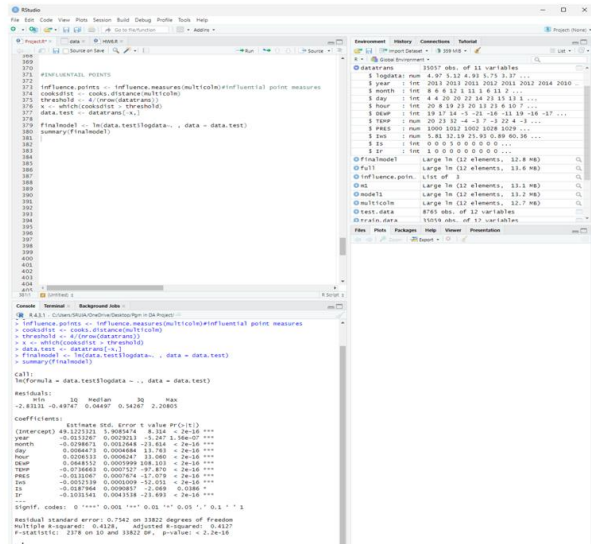


As we know when two independent variables are strongly correlated there is no need to keep both of them in the model! They don't add predictive value to the model. Therefore we can remove PRES variable from the model.



b) Influential points (removing)

Influential points are observations that significantly affect the fitted model; these are usually outliers. In the event that they are eliminated, the parameter estimations change



After removing influential points the Adjusted R-squared increased to: 0.4127

## EVALUATION STRATEGIES

### 5-Crossfold evaluation

RMSE -0.754288

RSquared - 0.4125214

MAE 0.6085027

The screenshot shows the RStudio interface with a script editor on the left, a console at the bottom, and the Environment pane on the right.

**Script Editor:**

```
384 #5-fold cross validation
385
386 install.packages("caret")
387 library(caret)
388
389 set.seed(123)
390 train.control <- trainControl(method = "cv", number=5) #train the model
391 model <- train(logdata, ~ data = data.test, method="lm", trControl=train.control)
392 print(model)$summary
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

**Console:**

```
R 6.4.1: C:\Users\Student\OneDrive\Documents\RStudio> RStudio --no-browser
> set.seed(123)
> train.control <- trainControl(method = "cv", number=5) #train the model
> model <- train(logdata, ~ data = data.test, method="lm", trControl=train.control)
> print(model)$summary
Linear regression
3813 samples
10 predictor
No pre-processing
Resampling: Cross-validated (5 fold)
Summary of sample sizes: 27067, 27067, 27067, 27067, 27067
Resampling results:
rmse      rsquared    mae
0.754288  0.4125214  0.6085027
Tuning parameter 'intercept' was held constant at a value of TRUE
>
```

**Environment:**

Object	Class	Attributes
data.test	data.frame	1813 obs. of 11 variables
data.train	data.frame	3507 obs. of 11 variables
final.model	lm	Large lm (12 elements, 12.8 MB)
logdata	data.frame	Large lm (12 elements, 12.8 MB)
influence.pot	matrix	Large matrix (12 elements, 10.4 MB)
lm	lm	Large lm (12 elements, 12.8 MB)
model	lm	Large lm (12 elements, 12.8 MB)
multicoll	matrix	Large matrix (12 elements, 12.8 MB)
test.data	data.frame	8765 obs. of 12 variables
train.data	data.frame	3507 obs. of 12 variables
values	matrix	Large matrix (3507 elements, 2.5 MB)
coefadj	matrix	Large matrix (3507 elements, 2.5 MB)
trf.val	matrix	Large matrix (3507 elements, 2.5 MB)
logdata	data.frame	Large matrix (3507 elements, 2.5 MB)
select.data	matrix	Large matrix (3507 elements, 2.5 MB)
threshold	matrix	Large matrix (3507 elements, 2.5 MB)
x	matrix	Large matrix (3507 elements, 2.5 MB)

## 5.2. Evaluations and Results

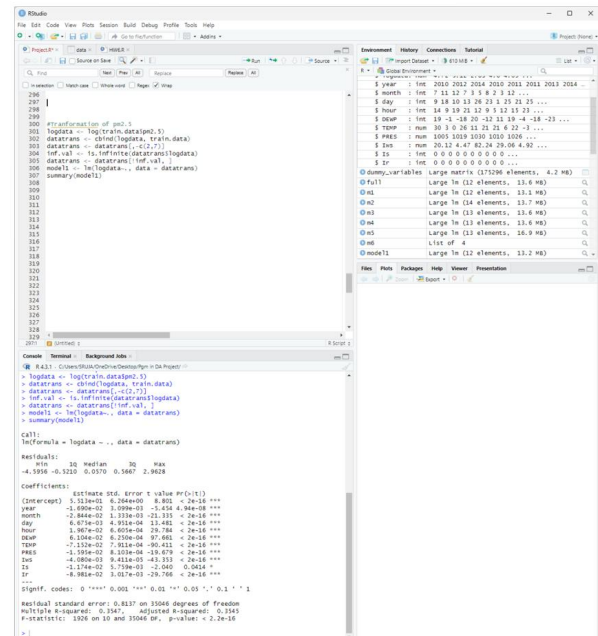
We concluded M2 is the best by looking at evaluation metrics RMSE and proceeded to improve the model further

### 3. POST PROCESSING

#### 3.1BEST MODEL

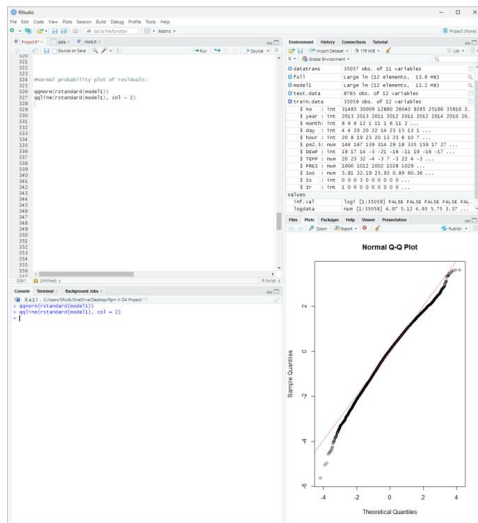
To Check Predictive performance we have considered Train.data in each model.

- Comparing RMSE of all the 5 Models.
- M1.Full Model- RMSE :77.36838
- M2.**Backward method using p-value in t-test as metric- RMSE :77.41664**
- M3.Backward method using AIC as metric: RMSE :77.36838
- M4.Forward method using AIC as metric: RMSE :77.36838
- M5.Stepwise method using AIC as metric: RMSE :77.36838
- Therefore M2 is Better Model compared to other models having better RMSE value.

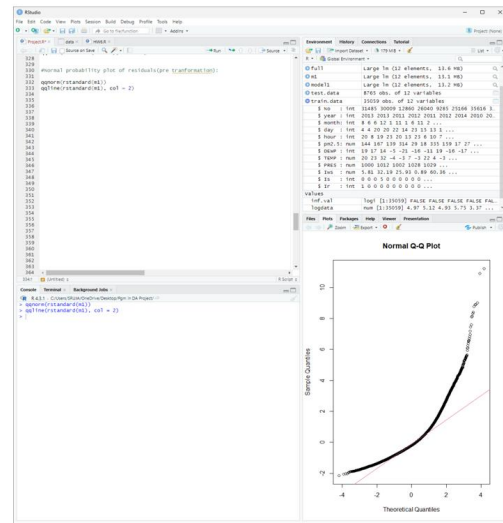


### 3.2 MODEL DIAGNOSIS

c) Normal probability plot of residuals: To check normality assumption for the error terms



QQ plot after Tranforming Pm2.5 variable.



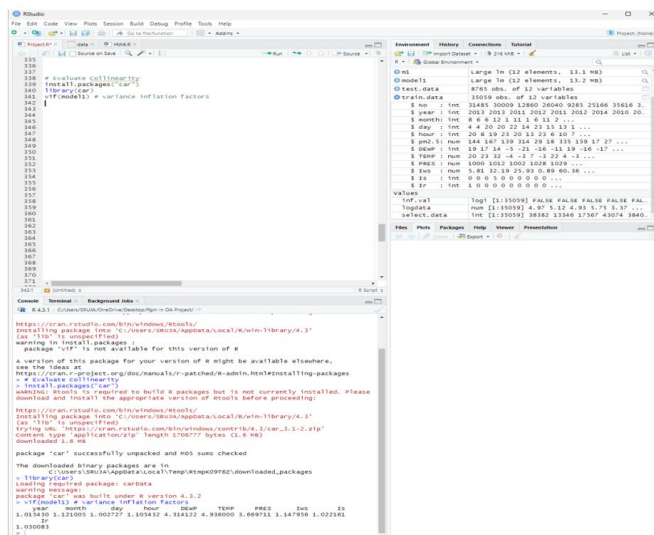
QQ plot before Tranforming Pm2.5 variable.

Looking at the QQ Plot for after Transforming PM2.5 variable there is a straight line which means the data is normally distributed

While Improving the model

a) Multicollinearity problem

Computing Variance Inflation Factor Statistics



Since VIF > 4 for DEWP ,TEMP, we are testing corr to find pair of independent variables

### 5.3. Findings

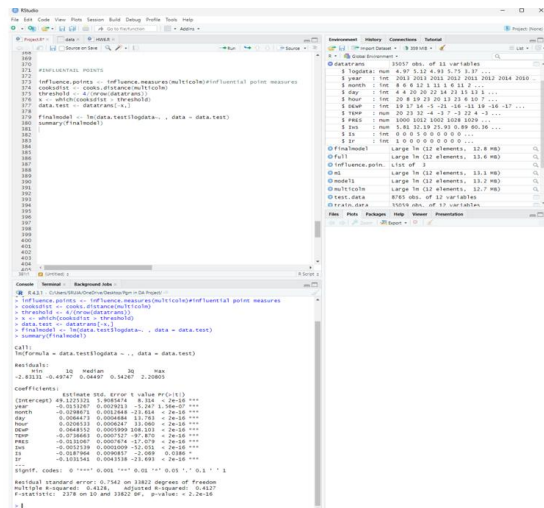
The best model we could offer for this data set

## Applying 5 regression models

Selecting the best model using best RMSE value and Adj R2 value.

Doing Residual analysis and fixing problem by transforming Y variable which is PM2.5

## Improving the model checking multi collinearity problem and dropping inferential points



After removing influential points the Adjusted R-squared increased to: 0.4127

ALL THE regression model gives below average accuracy even after transformation and Model improvements techniques like Multicollinearity, dropping Inferential points and dealing with missing values.

## 6. Conclusions and Future Work

### 6.1. Conclusions

We have performed multiple tests and analyzed the data set at various levels. We can conclude that although all the approaches or tests show relatively less accuracy, ALL THE regression model gives below average accuracy even after transformation and Model improvements techniques like Multicollinearity, dropping Inferential points and dealing with missing values. Also, the number of data points is very small, and hence, an increase in the data volume with respect to the number of rows may make the analysis easier and more meaningful with better accuracy. We might also learn and apply different approach to tackle such datasets.

### 6.2. Limitations

- We need to find out more approaches to improve the model. Domain knowledge is lacking
- Data Quality
- Model complexity
- Correlation of Data
- Feature Quality

## 6.3. Potential Improvements or Future Work

### 1. Data Pre Processing

- Better Data Cleaning can help
- To drop nominal variable
- Cross Validation strategies
- Continuous learning to best fit the data to use in the model

### 2. Linear regression model

Increase domain knowledge of various Linear regression model to implement it to the right data set

### 3. Post processing

Adapt enhanced Post processing techniques like better smoothing techniques, calibration etc.,.