

```

1 from google.colab import files
2
3 uploaded = files.upload()
4
5 for fn in uploaded.keys():
6     print('User uploaded file "{name}" with length {length} bytes'.format(
7         name=fn, length=len(uploaded[fn])))

```



Choose Files 3 files

- **sample_submission.csv**(application/vnd.ms-excel) - 31939 bytes, last modified: 11/28/2018 - 100% d
- **test.csv**(application/vnd.ms-excel) - 451405 bytes, last modified: 11/28/2018 - 100% done
- **train.csv**(application/vnd.ms-excel) - 460676 bytes, last modified: 11/28/2018 - 100% done

Saving sample_submission.csv to sample_submission.csv
 Saving test.csv to test.csv
 Saving train.csv to train.csv
 User uploaded file "sample_submission.csv" with length 31939 bytes
 User uploaded file "test.csv" with length 451405 bytes
 User uploaded file "train.csv" with length 460676 bytes

```

1 from google.colab import drive
2 drive.mount('/content/gdrive')
3 root_path = 'gdrive/My Drive/Data_From_colab'

```



Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=9473

Enter your authorization code:

.....

Mounted at /content/gdrive

```

1 import pandas as pd
2 import numpy as np
3 from sklearn import model_selection
4 import lightgbm as lgb
5 from sklearn.model_selection import KFold
6 from sklearn.metrics import mean_squared_error
7 from IPython.display import clear_output
8 from sklearn.model_selection import cross_val_score
9
10 train = pd.read_csv("train.csv")
11 test = pd.read_csv("test.csv")
12
13 test.head()

```



	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandCon
0	1461	20	RH	80.0	11622	Pave	NaN	Reg	
1	1462	20	RL	81.0	14267	Pave	NaN	IR1	
2	1463	60	RL	74.0	13830	Pave	NaN	IR1	
3	1464	60	RL	78.0	9978	Pave	NaN	IR1	
4	1465	120	RL	43.0	5005	Pave	NaN	IR1	

```

1 #id
2 test_id = test["Id"]
3
4 train = train.drop("Id",axis=1)
5 test = test.drop("Id",axis=1)

```

```
1 train.head()
```



	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	
0	60	RL	65.0	8450	Pave	NaN	Reg		Lvl
1	20	RL	80.0	9600	Pave	NaN	Reg		Lvl
2	60	RL	68.0	11250	Pave	NaN	IR1		Lvl
3	70	RL	60.0	9550	Pave	NaN	IR1		Lvl
4	60	RL	84.0	14260	Pave	NaN	IR1		Lvl

```

1 # poolQC
2 train["PoolQC"] = train["PoolQC"].fillna("None")
3 test["PoolQC"] = test["PoolQC"].fillna("None")
4
5
6 #MiscFeature
7 train["MiscFeature"] = train["MiscFeature"].fillna("None")
8 test["MiscFeature"] = test["MiscFeature"].fillna("None")
9
10 #Alley
11 train["Alley"] = train["Alley"].fillna("None")
12 test["Alley"] = test["Alley"].fillna("None")
13
14 #Fence
15 train["Fence"] = train["Fence"].fillna("None")
16 test["Fence"] = test["Fence"].fillna("None")
17
18 #fireplace
19 train["FireplaceQu"] = train["FireplaceQu"].fillna("None")
20 test["FireplaceQu"] = test["FireplaceQu"].fillna("None")
21
22 #lotfrontage
23 train["LotFrontage"] = train.groupby("Neighborhood")["LotFrontage"].transform(lambda x: x.fillna(x.median()))
24 test["LotFrontage"] = test.groupby("Neighborhood")["LotFrontage"].transform(lambda x: x.fillna(x.median()))
25
26 #garage
27 for col in ('GarageType', 'GarageFinish', 'GarageQual', 'GarageCond'):
28     train[col] = train[col].fillna('None')
29     test[col] = test[col].fillna('None')
30 for col in ('GarageYrBlt', 'GarageArea', 'GarageCars'):
31     train[col] = train[col].fillna(0)
32     test[col] = test[col].fillna(0)
33 del col
34 # bsmtfin
35 for col in ('BsmtFinSF1', 'BsmtFinSF2', 'BsmtUnfSF', 'TotalBsmtSF', 'BsmtFullBath', 'BsmtHalfBath'):
36     train[col] = train[col].fillna(0)
37     test[col] = test[col].fillna(0)
38

```

```

39 for col in ("BsmtQual", 'BsmtCond', 'BsmtExposure', 'BsmtFinType1', 'BsmtFinType2'):
40     train[col] = train[col].fillna("None")
41     test[col] = test[col].fillna("None")
42 del col
43
44 # masvnr
45 train["MasVnrType"] = train["MasVnrType"].fillna("None")
46 test["MasVnrType"] = test["MasVnrType"].fillna("None")
47
48 train["MasVnrArea"] = train["MasVnrArea"].fillna(0)
49 test["MasVnrArea"] = test["MasVnrArea"].fillna(0)
50
51 #MSZoning
52 train['MSZoning'] = train['MSZoning'].fillna(train['MSZoning'].mode()[0])
53 test['MSZoning'] = test['MSZoning'].fillna(test['MSZoning'].mode()[0])
54
55 #Utilities
56 train = train.drop(['Utilities'], axis=1)
57 test = test.drop(['Utilities'], axis=1)
58
59 #Functional
60 train["Functional"] = train["Functional"].fillna("Typ")
61 test["Functional"] = test["Functional"].fillna("Typ")
62
63 #Electrical
64 train["Electrical"] = train["Electrical"].fillna("SBrkr")
65 test["Electrical"] = test["Electrical"].fillna("SBrkr")
66
67 # Kitchen
68 train['KitchenQual'] = train['KitchenQual'].fillna(train['KitchenQual'].mode()[0])
69 test['KitchenQual'] = test['KitchenQual'].fillna(test['KitchenQual'].mode()[0])
70
71 #Exterior
72 train['Exterior1st'] = train['Exterior1st'].fillna(train['Exterior1st'].mode()[0])
73 test['Exterior1st'] = test['Exterior1st'].fillna(test['Exterior1st'].mode()[0])
74
75 train['Exterior2nd'] = train['Exterior2nd'].fillna(train['Exterior2nd'].mode()[0])
76 test['Exterior2nd'] = test['Exterior2nd'].fillna(test['Exterior2nd'].mode()[0])
77
78 #saletype
79 train['SaleType'] = train['SaleType'].fillna(train['SaleType'].mode()[0])
80 test['SaleType'] = test['SaleType'].fillna(test['SaleType'].mode()[0])
81 #Mssubclass
82 train['MSSubClass'] = train['MSSubClass'].fillna("None")
83 test['MSSubClass'] = test['MSSubClass'].fillna("None")
84
85 train['MSSubClass'] = train['MSSubClass'].apply(str)
86 test['MSSubClass'] = test['MSSubClass'].apply(str)
87
88 #overallcond
89 train['OverallCond'] = train['OverallCond'].astype(str)
90 test['OverallCond'] = test['OverallCond'].astype(str)
91
92 #yrsold mosold
93 train['YrSold'] = train['YrSold'].astype(str)
94 test['YrSold'] = test['YrSold'].astype(str)
95
96 train['MoSold'] = train['MoSold'].astype(str)
97 test['MoSold'] = test['MoSold'].astype(str)
98
99 # year
100 train['YrBltAndRemod'] = train['YearBuilt'] + train['YearRemodAdd']
101 test['YrBltAndRemod'] = test['YearBuilt'] + test['YearRemodAdd']
102
103 #total
104 train['TotalSF'] = train['TotalBsmtSF'] + train['1stFlrSF'] + train['2ndFlrSF']
105 test['TotalSF'] = test['TotalBsmtSF'] + test['1stFlrSF'] + test['2ndFlrSF']
106
107 train['Total_sqr_footage'] = (train['BsmtFinSF1'] + train['BsmtFinSF2'] + train['1stFlrSF'] +
108 test['Total_sqr_footage'] = (test['BsmtFinSF1'] + test['BsmtFinSF2'] + test['1stFlrSF'] + test['2ndFlrSF'])
109
110 train['Total_Bathrooms'] = (train['FullBath'] + (0.5 * train['HalfBath']) + train['BsmtFullBath'] +
111 test['Total_Bathrooms'] = (test['FullBath'] + (0.5 * test['HalfBath']) + test['BsmtFullBath'] +
112

```

```

113 train['Total_porch_sf'] = (train['OpenPorchSF'] + train['3SsnPorch'] + train['EnclosedPorch'])
114 test['Total_porch_sf'] = (test['OpenPorchSF'] + test['3SsnPorch'] + test['EnclosedPorch'])
115
116 #street
117 train = train.drop('Street',axis=1)
118 test = test.drop('Street',axis=1)
119
120
121 train = train.fillna(train.median())
122 test = test.fillna(test.median())
123
124 #train = train.sample(frac=1, random_state=0)
125
126 co_box = []
127 for co in train.columns:
128     try:
129         sumup = train[co].sum()
130         if(type(sumup) == type("dokabenman")):
131             co_box.append(co)
132     except:
133         print(co)
134
135 from sklearn.preprocessing import LabelEncoder
136 for obj_col in co_box:
137     le = LabelEncoder()
138     train[obj_col] = train[obj_col].apply(lambda x:str(x))
139     train[obj_col] = pd.DataFrame({obj_col:le.fit_transform(train[obj_col])})
140
141     test[obj_col] = test[obj_col].apply(lambda x:str(x))
142     test[obj_col] = pd.DataFrame({obj_col:le.fit_transform(test[obj_col])})
143
144
145 train["SalePrice"] = np.log(train["SalePrice"])

```

```
1 train.head()
```

	MSSubClass	MSZoning	LotFrontage	LotArea	Alley	LotShape	LandContour	LotConfig
0	9	3	65.0	8450	1	3	3	1
1	4	3	80.0	9600	1	3	3	1
2	9	3	68.0	11250	1	0	3	1
3	10	3	60.0	9550	1	0	3	1
4	9	3	84.0	14260	1	0	3	1

5 rows × 83 columns

```

1 pur,exa = train["SalePrice"].copy(),train.drop("SalePrice",axis=1).copy()
2 pur = (pur - pur.mean())/pur.std()
3 kf = KFold(n_splits=3, shuffle=True, random_state=124)
4 model =lgb.LGBMRegressor(random_state=0,boosting_type="gbdt",objective="regression",metric="r
5 result = []
6 for tr ,te in kf.split(pur):
7     X_train, y_train = exa.iloc[tr], pur.iloc[tr]
8     X_test, y_test = exa.iloc[te], pur.iloc[te]
9     model.fit(X_train,y_train)
10    pre = model.predict(X_test)
11    result.append(mean_squared_error(y_test,pre))
12 print(result)
13 print(np.mean(result))

```



```

/usr/local/lib/python3.6/dist-packages/lightgbm/engine.py:118: UserWarning: Found `nu
warnings.warn("Found `{}` in params. Will use it instead of argument".format(alias)
/usr/local/lib/python3.6/dist-packages/lightgbm/engine.py:118: UserWarning: Found `nu
warnings.warn("Found `{}` in params. Will use it instead of argument".format(alias)
/usr/local/lib/python3.6/dist-packages/lightgbm/engine.py:118: UserWarning: Found `nu
warnings.warn("Found `{}` in params. Will use it instead of argument".format(alias)

```

```

1 pur,exa = train["SalePrice"].copy(),train.drop("SalePrice",axis=1).copy()
2 mean = pur.mean()
3 std = pur.std()
4 pur = (pur - mean) / std
5 model =lgb.LGBMRegressor(random_state=0,boosting_type="gbdt",objective="regression",metric="r
6 model.fit(exa,pur)
7 pre = model.predict(test)
8 pre = pre*std + mean
9 pre = np.exp(pre)
10
11 submission = pd.DataFrame({
12     "Id": test_id,
13     "SalePrice": pre
14 })
15 submission.to_csv('gdrive/My Drive/Data_From_colab/aiwp_task3.csv', index=False)

```

```

➡ /usr/local/lib/python3.6/dist-packages/lightgbm/engine.py:118: UserWarning: Found `nu
warnings.warn("Found `{}` in params. Will use it instead of argument".format(alias)

```

1

House IDs and Predictions:		1490	202260.1587	1521	137913.4503
Id	SalePrice	1491	184926.3286	1522	159551.6561
1461	124910.35	1492	98235.14793	1523	114829.9371
1462	160076.5745	1493	183123.432	1524	125729.572
1463	170292.1525	1494	292168.0526	1525	124049.4615
1464	189264.4782	1495	292784.6219	1526	115170.2534
1465	185506.8482	1496	238210.2583	1527	96295.89897
1466	177417.2436	1497	191675.9605	1528	125119.5101
1467	171166.6832	1498	154780.4832	1529	146031.7515
1468	171713.7272	1499	154425.9836	1530	159847.6079
1469	181636.6352	1500	145921.6357	1531	108182.5365
1470	119939.08	1501	175058.7865	1532	99816.27471
1471	208650.0985	1502	156294.8798	1533	136184.3133
1472	100981.9685	1503	317779.0332	1534	118343.1373
1473	94775.35285	1504	238252.9783	1535	157701.8673
1474	148427.5708	1505	223280.9826	1536	115919.4607
1475	142470.8166	1506	177267.3875	1537	64399.20918
1476	350599.1982	1507	226126.0951	1538	164320.7635
1477	255989.0333	1508	217202.9797	1539	169064.6731
1478	284569.8651	1509	164482.4291	1540	94744.69565
1479	253254.7413	1510	158307.9953	1541	135360.3328
1480	487410.8642	1511	149800.3933	1542	150950.074
1481	332413.8953	1512	166722.7481	1543	235562.0721
1482	205412.7739	1513	147848.3552	1544	74164.40033
1483	183765.4445	1514	180971.868	1545	111956.2192
1484	176305.7497	1515	176931.7461	1546	131419.7155
1485	176607.2328	1516	156230.5312	1547	131786.5072
1486	193709.2559	1517	151372.4257	1548	137271.7656
1487	337017.0661	1518	118639.4414	1549	113513.3711
1488	229032.9133	1519	225003.946	1550	138210.4829
1489	201935.7211	1520	133616.2871	1551	113533.178

1552	124919.5239	1583	333408.1271	1614	117208.9297
1553	151170.478	1584	230527.3105	1615	81706.22258
1554	103787.3766	1585	139620.1864	1616	71554.21552
1555	174133.0514	1586	62544.4942	1617	91661.42568
1556	72811.49781	1587	99428.28107	1618	129363.3806
1557	114046.059	1588	149420.7904	1619	132790.3894
1558	85534.28638	1589	93176.35457	1620	156475.9575
1559	91152.40538	1590	132104.4492	1621	146788.2479
1560	129469.4913	1591	99614.77806	1622	142558.3097
1561	152855.4237	1592	120592.8089	1623	313167.0195
1562	123251.3264	1593	110571.1739	1624	198124.7484
1563	115226.4968	1594	158401.4709	1625	114574.2096
1564	159917.5789	1595	115747.3669	1626	162822.685
1565	154921.1237	1596	251239.3492	1627	186018.5461
1566	212438.9348	1597	183524.3047	1628	280293.88
1567	70701.0193	1598	183948.8743	1629	173216.9598
1568	245315.9381	1599	158522.0139	1630	385718.6158
1569	165987.5815	1600	160855.4489	1631	215734.5053
1570	139151.4856	1601	69463.04297	1632	245921.316
1571	115879.6712	1602	117124.8756	1633	174214.1479
1572	147393.8929	1603	83157.67547	1634	179362.1578
1573	245452.9237	1604	283479.9699	1635	171803.6661
1574	105734.2522	1605	242365.9544	1636	152880.7874
1575	211036.6551	1606	167801.4645	1637	189636.347
1576	225723.1596	1607	145776.4783	1638	189859.2253
1577	195071.2796	1608	226758.7586	1639	176766.3081
1578	145726.9249	1609	195404.5493	1640	259958.0727
1579	139100.3993	1610	163308.2108	1641	173317.4063
1580	188200.2861	1611	141544.1723	1642	229767.8405
1581	138600.6509	1612	157463	1643	216795.6064
1582	132268.1092	1613	165781.183	1644	233252.6214

1645	193126.4562	1676	192085.0988	1707	213126.9735
1646	160571.4086	1677	220842.8886	1708	223581.9615
1647	163949.9879	1678	455559.4439	1709	267233.0432
1648	139272.4348	1679	399347.3404	1710	226230.5179
1649	139694.9443	1680	335933.2849	1711	250485.8491
1650	109488.7113	1681	234272.1515	1712	257555.1132
1651	115209.2195	1682	307549.0412	1713	267300.5658
1652	97187.67172	1683	178553.7214	1714	229586.9288
1653	97976.59458	1684	170804.3547	1715	202342.1588
1654	150895.9507	1685	173757.475	1716	183883.618
1655	152183.5409	1686	165240.3011	1717	164719.804
1656	149674.4323	1687	169305.764	1718	137120.6588
1657	152501.8536	1688	187754.6681	1719	203195.0667
1658	149696.9085	1689	188711.5795	1720	261093.9822
1659	148964.7208	1690	184509.1827	1721	166098.0798
1660	146472.4984	1691	181653.8638	1722	124620.5536
1661	543481.7281	1692	246696.4977	1723	153594.0211
1662	393034.4012	1693	169366.2482	1724	227055.6543
1663	368719.4619	1694	188465.3607	1725	241645.2507
1664	540544.4389	1695	170617.7763	1726	187700.8964
1665	303973.5799	1696	255130.7568	1727	163919.9297
1666	335483.3978	1697	173502.2777	1728	178779.8634
1667	365858.67	1698	391344.6788	1729	165841.6679
1668	320183.2713	1699	342615.2795	1730	149029.7848
1669	303430.0697	1700	234818.411	1731	122738.078
1670	333375.9482	1701	269626.1138	1732	132815.7723
1671	268261.6364	1702	251421.6186	1733	114101.3446
1672	539775.1484	1703	259275.6439	1734	118774.5994
1673	284650.8587	1704	301686.6621	1735	132757.7187
1674	240222.1789	1705	219619.8828	1736	122621.0673
1675	194870.4903	1706	487758.1852	1737	351570.3189

1738	227406.3531	1769	170858.8609	1800	122892.055
1739	280231.218	1770	143745.5075	1801	133537.0401
1740	200090.0073	1771	127806.7653	1802	108050.6256
1741	189840.4071	1772	148531.9935	1803	151212.6485
1742	172673.7162	1773	129512.6454	1804	140089.9757
1743	170876.1345	1774	113184.433	1805	137804.5009
1744	291104.7912	1775	134305.8175	1806	118219.8163
1745	235962.0704	1776	126408.3942	1807	129489.3951
1746	197688.5346	1777	118852.5188	1808	110849.1847
1747	211439.6362	1778	134583.8897	1809	126865.5618
1748	224034.457	1779	112874.3844	1810	141171.7854
1749	154953.4787	1780	170194.8075	1811	103509.2025
1750	134907.7526	1781	132548.0155	1812	99158.93625
1751	234423.8663	1782	82414.85645	1813	129895.2149
1752	120523.905	1783	157194.5891	1814	91435.04558
1753	141758.1068	1784	100076.4355	1815	52109.19081
1754	178151.3441	1785	116597.6868	1816	97149.24455
1755	175037.0222	1786	129056.0241	1817	102988.2892
1756	127260.7352	1787	154532.72	1818	129944.1923
1757	121011.2891	1788	53921.32523	1819	112845.2409
1758	149266.4408	1789	81309.80154	1820	68979.37018
1759	155015.1931	1790	74860.04669	1821	98526.31392
1760	164089.549	1791	165970.7273	1822	145163.7585
1761	142039.7348	1792	156672.9145	1823	56575.402
1762	179349.0604	1793	132375.9475	1824	136957.5044
1763	171112.458	1794	141464.1968	1825	131435.2062
1764	114965.7603	1795	133542.8516	1826	101191.5524
1765	166638.2313	1796	110193.1698	1827	97888.3251
1766	171651.9658	1797	120017.9196	1828	135257.5252
1767	212345.57	1798	127024.5067	1829	123243.9789
1768	145140.7584	1799	122181.1422	1830	136890.9243

1831	139901.162	1862	316022.7099	1893	126573.0416
1832	94058.36225	1863	315755.2516	1894	123271.1831
1833	140352.1937	1864	315755.2516	1895	140647.3615
1834	119785.897	1865	321255.8153	1896	125739.4059
1835	144421.6551	1866	325972.7872	1897	108243.2348
1836	130154.5061	1867	248590.0913	1898	105766.3782
1837	93244.98644	1868	296575.6911	1899	141371.8935
1838	130105.7813	1869	218852.1212	1900	158469.6922
1839	79126.70004	1870	226914.4267	1901	166999.3941
1840	165229.8936	1871	233638.748	1902	127702.7456
1841	151571.6309	1872	170510.5755	1903	224359.3077
1842	83377.50802	1873	252366.9963	1904	135351.6559
1843	125384.5257	1874	139869.9294	1905	198642.0958
1844	140269.1215	1875	211257.0253	1906	176681.8538
1845	139568.7626	1876	196227.9764	1907	223440.3475
1846	155696.8201	1877	211840.0697	1908	109605.4401
1847	161565.962	1878	198029.7174	1909	136162.2378
1848	56809.96249	1879	126442.9026	1910	115560.1055
1849	111205.6631	1880	137589.4048	1911	197027.3011
1850	113785.6912	1881	242374.6872	1912	299804.7498
1851	155773.4381	1882	246788.4397	1913	136672.7591
1852	129429.602	1883	193198.3928	1914	63377.98408
1853	125530.2368	1884	208851.0747	1915	330681.7726
1854	170898.2242	1885	233981.8432	1916	64492.50118
1855	138913.3836	1886	279978.5227	1917	254318.6264
1856	223028.5094	1887	217523.8284	1918	140901.7092
1857	138398.5161	1888	300378.5883	1919	171589.3802
1858	140391.6919	1889	158559.6954	1920	147197.303
1859	116082.1088	1890	130998.6696	1921	459201.3795
1860	143700.1144	1891	142509.9951	1922	315313.2444
1861	115709.1615	1892	102712.3376	1923	218014.5544

1924	262621.1109	1955	137620.5697	1986	210101.3098
1925	203399.659	1956	306488.6582	1987	171649.6497
1926	390468.7103	1957	144488.874	1988	179776.854
1927	131385.0329	1958	311212.574	1989	189861.939
1928	156064.7713	1959	140027.9603	1990	207812.5156
1929	128251.0514	1960	112990.6034	1991	240138.7063
1930	126558.7198	1961	122727.2743	1992	220718.9035
1931	139470.9168	1962	97797.40471	1993	175448.4023
1932	145494.2365	1963	106083.8644	1994	237110.5708
1933	178081.2497	1964	117162.8082	1995	184038.6586
1934	180221.7312	1965	150681.0918	1996	240574.3968
1935	175316.6991	1966	151315.6915	1997	281834.669
1936	187181.885	1967	299624.8293	1998	350065.9364
1937	184771.7945	1968	482140.7444	1999	254094.4217
1938	156809.279	1969	367600.9586	2000	303878.062
1939	240396.6509	1970	496102.4496	2001	267779.2922
1940	179727.146	1971	474721.1454	2002	242425.9929
1941	175792.035	1972	375668.2917	2003	275169.257
1942	187318.0851	1973	269461.9647	2004	268217.7101
1943	193366.119	1974	334540.1	2005	219502.6925
1944	331024.4573	1975	529425.5931	2006	238420.224
1945	372169.3007	1976	266551.7037	2007	248382.7907
1946	135708.366	1977	350209.0939	2008	200270.8153
1947	325330.0072	1978	350556.8641	2009	194783.5736
1948	170646.6455	1979	300527.5679	2010	188112.1774
1949	246988.4269	1980	198020.0478	2011	141911.6524
1950	177408.3724	1981	333143.7228	2012	173002.9486
1951	240182.4011	1982	230079.5466	2013	183440.5564
1952	220833.0168	1983	201768.987	2014	185467.672
1953	178386.3318	1984	168864.0198	2015	197761.8355
1954	178749.0314	1985	212170.326	2016	196126.562

2017	194754.4666	2048	139065.8363	2079	134202.62
2018	120863.821	2049	131101.7319	2080	114677.0914
2019	127839.3903	2050	182125.513	2081	124440.5248
2020	111404.3272	2051	103824.534	2082	127702.1846
2021	106764.6674	2052	112462.4747	2083	141131.7496
2022	183344.0267	2053	142783.7885	2084	109458.6703
2023	159019.1928	2054	81692.82183	2085	118174.4814
2024	233005.0037	2055	147292.5465	2086	100365.447
2025	375772.248	2056	137570.5129	2087	124515.6053
2026	171088.9954	2057	115068.7387	2088	116493.5243
2027	143922.258	2058	231610.3726	2089	78214.55173
2028	149680.7139	2059	135358.7114	2090	135385.3712
2029	176071.5373	2060	172785.9265	2091	130228.8029
2030	273579.0051	2061	164467.5475	2092	129105.4837
2031	213974.8823	2062	129155.1346	2093	123064.6127
2032	239268.1385	2063	119215.9258	2094	123276.0822
2033	259083.1569	2064	139189.5522	2095	155113.7493
2034	163892.3868	2065	131332.1469	2096	81922.61976
2035	215499.5045	2066	161365.2779	2097	94682.8885
2036	181227.0639	2067	108729.9302	2098	143616.9144
2037	186135.303	2068	117047.9839	2099	59864.84775
2038	278663.895	2069	85813.32413	2100	99094.03946
2039	208905.9089	2070	123787.5844	2101	132887.8564
2040	333817.0287	2071	85366.40628	2102	134270.6736
2041	307000.5567	2072	135729.3352	2103	111122.3162
2042	199030.9967	2073	138599.0626	2104	130598.0056
2043	174322.3585	2074	176681.633	2105	125863.2969
2044	150648.9403	2075	138917.108	2106	62305.82033
2045	234662.2943	2076	109218.2428	2107	212243.6615
2046	134934.6441	2077	163236.551	2108	107819.005
2047	151390.0452	2078	130515.1366	2109	114466.725

2110	136891.1	2141	147538.9079	2172	126289.9908
2111	137573.6697	2142	123874.1951	2173	158301.0336
2112	147564.8624	2143	150421.0203	2174	225663.1112
2113	120296.8184	2144	112523.0414	2175	284084.2332
2114	106369.2022	2145	135106.9231	2176	336269.8837
2115	159584.7927	2146	159804.1624	2177	238606.1973
2116	116759.5724	2147	160628.5379	2178	206613.8133
2117	155804.6104	2148	141859.5659	2179	138648.2804
2118	119001.2948	2149	140445.4529	2180	220957.6431
2119	101310.1987	2150	252989.8627	2181	186605.2878
2120	107243.1586	2151	141082.4511	2182	219703.1383
2121	87704.45056	2152	186143.2892	2183	203949.5648
2122	108365.6088	2153	161353.4376	2184	126885.1377
2123	85184.89202	2154	112437.0108	2185	130312.8048
2124	162076.2298	2155	142650.7036	2186	146025.1503
2125	118530.1554	2156	233128.2489	2187	153285.0158
2126	162358.5974	2157	225572.1276	2188	162069.601
2127	132961.5904	2158	244577.074	2189	201882.227
2128	130297.4803	2159	200319.6892	2190	81830.87047
2129	90772.15422	2160	176899.3401	2191	83165.55328
2130	142732.1835	2161	247129.0902	2192	107060.0241
2131	130245.059	2162	398401.0203	2193	100260.8538
2132	115845.1324	2163	347163.3799	2194	109388.7932
2133	132144.5374	2164	246860.9414	2195	108047.0837
2134	117216.8078	2165	205856.4105	2196	96278.70244
2135	93914.17371	2166	144847.222	2197	120278.4205
2136	71162.09018	2167	224656.4944	2198	180052.9177
2137	107532.413	2168	209079.8854	2199	179215.5158
2138	136008.1287	2169	190639.2648	2200	149014.2565
2139	150258.8559	2170	224526.0668	2201	142191.843
2140	135268.8073	2171	145117.3275	2202	182271.387

2203	146041.3143	2234	234150.1855	2265	161153.0294
2204	201774.9971	2235	202474.5275	2266	300185.8286
2205	102049.3605	2236	273538.8785	2267	332650.7136
2206	136378.1082	2237	335500.1718	2268	397254.6843
2207	191307.7247	2238	203380.3366	2269	155447.7474
2208	221945.0319	2239	104246.1898	2270	184356.9517
2209	219098.7209	2240	159036.2126	2271	223276.3847
2210	123573.3795	2241	155560.1535	2272	199981.1724
2211	110304.6834	2242	126022.3064	2273	159078.9626
2212	113662.4795	2243	119752.6942	2274	178732.798
2213	111134.4636	2244	106844.0879	2275	162754.7024
2214	135398.1687	2245	95939.84889	2276	198386.6091
2215	105260.1955	2246	140760.6278	2277	194424.9894
2216	139563.954	2247	105537.4649	2278	141838.8251
2217	59296.11602	2248	125503.7421	2279	123934.8279
2218	84653.14171	2249	125275.4104	2280	119605.3428
2219	69755.06901	2250	132254.7976	2281	178054.0617
2220	72017.48463	2251	111106.3412	2282	187111.8074
2221	329359.5192	2252	168139.311	2283	101025.7132
2222	260226.8237	2253	148527.044	2284	111004.3653
2223	279357.3846	2254	171197.8966	2285	154141.5605
2224	208735.1134	2255	190428.0036	2286	125469.9558
2225	130466.5896	2256	177575.3986	2287	347181.4761
2226	185014.0357	2257	213730.4351	2288	261003.6627
2227	211986.6508	2258	165053.6942	2289	424266.7354
2228	288610.4762	2259	175954.2645	2290	470304.1322
2229	260355.2347	2260	150000.8346	2291	308996.1761
2230	164321.7493	2261	177504.7581	2292	413771.6752
2231	212655.4306	2262	200023.7076	2293	481294.7794
2232	173008.8984	2263	387951.886	2294	446764.0718
2233	184531.3947	2264	484582.0168	2295	491179.5756

2296	274052.0569	2327	200632.7202	2358	196285.9637
2297	314517.8267	2328	223862.9686	2359	132679.6172
2298	312907.6468	2329	188247.6356	2360	117906.3457
2299	347550.4681	2330	176877.5419	2361	145201.8044
2300	351143.8935	2331	369191.1771	2362	287090.0091
2301	271131.2914	2332	428599.5391	2363	133437.8088
2302	234161.0043	2333	333298.2499	2364	149722.3662
2303	236729.4595	2334	240520.3382	2365	217871.1493
2304	247585.8803	2335	264538.0849	2366	179445.91
2305	186065.3982	2336	345247.4922	2367	226746.9108
2306	186465.8822	2337	221312.0431	2368	221671.7587
2307	192385.525	2338	286691.5165	2369	229802.634
2308	225966.6524	2339	225476.5668	2370	172456.0292
2309	277502.1905	2340	362835.7683	2371	173980.7504
2310	201920.4212	2341	246210.8413	2372	192320.0961
2311	199954.7385	2342	259933.0254	2373	300306.2802
2312	178089.2224	2343	235364.2529	2374	314840.6091
2313	178884.2581	2344	221776.371	2375	250485.3655
2314	164340.3229	2345	245719.7009	2376	302035.3057
2315	175763.4869	2346	225375.138	2377	363130.0367
2316	177981.1689	2347	214253.2004	2378	142938.2183
2317	185366.7975	2348	250101.6978	2379	211292.9643
2318	174154.3744	2349	170713.2626	2380	149612.7136
2319	175761.8886	2350	269317.2593	2381	168438.7116
2320	180174.8571	2351	261252.3156	2382	186948.3574
2321	243407.583	2352	259621.6631	2383	211117.1648
2322	160742.6409	2353	220726.9206	2384	236762.3639
2323	176924.6332	2354	138696.1267	2385	162313.0737
2324	175737.719	2355	142625.5219	2386	137953.2201
2325	214310.381	2356	147903.2169	2387	137561.7655
2326	174513.1448	2357	199311.7327	2388	91381.36282

2389	121802.2055	2420	120509.0236	2451	122231.8555
2390	143258.5926	2421	157107.2882	2452	209580.2
2391	140413.1032	2422	115726.6118	2453	100904.0845
2392	115969.7622	2423	117401.9649	2454	124198.3311
2393	173730.2471	2424	141493.0043	2455	123573.6674
2394	149189.5723	2425	197456.7512	2456	137152.3887
2395	200813.5742	2426	130054.4395	2457	131836.6884
2396	156050.287	2427	128113.1802	2458	118167.2329
2397	205802.2958	2428	169110.0603	2459	114575.9199
2398	125520.4887	2429	114561.2499	2460	139963.6691
2399	55161.65331	2430	132687.1735	2461	126369.4267
2400	57277.62959	2431	109743.2475	2462	141927.0293
2401	119475.8807	2432	136836.2611	2463	123097.2778
2402	135996.4869	2433	141570.6653	2464	153413.3387
2403	131841.9699	2434	131871.9301	2465	140686.3563
2404	150889.8279	2435	157714.3299	2466	116792.5618
2405	164470.7774	2436	108828.4886	2467	152393.9045
2406	144857.5392	2437	110204.911	2468	78066.62926
2407	122834.044	2438	129352.7214	2469	75078.60519
2408	139526.4028	2439	110844.8085	2470	189837.6077
2409	124231.6603	2440	132245.1626	2471	183134.3602
2410	157859.3774	2441	98828.30877	2472	158765.3012
2411	120640.3743	2442	89118.41745	2473	123818.2301
2412	163944.1176	2443	121538.7611	2474	86677.46703
2413	129762.1457	2444	131637.5097	2475	230521.3827
2414	134561.1556	2445	93674.46372	2476	127949.1889
2415	118247.99	2446	122184.2575	2477	131306.9481
2416	125193.8708	2447	166945.9984	2478	143544.5711
2417	133595.1999	2448	130441.1803	2479	104717.4572
2418	127916.5794	2449	115197.6535	2480	149352.4981
2419	131981.1888	2450	149354.1721	2481	119585.2038

2482	131660.616	2513	218799.1749	2544	126368.0999
2483	115293.4308	2514	233390.3926	2545	134771.3125
2484	126633.5904	2515	144907.3133	2546	130461.0912
2485	129057.5268	2516	168226.0273	2547	135759.5898
2486	156142.4556	2517	136225.9041	2548	161103.4787
2487	173505.3032	2518	142816.3996	2549	160485.4935
2488	169372.0629	2519	223935.3145	2550	288208.7387
2489	154103.1609	2520	210191.7818	2551	140378.589
2490	152308.9378	2521	193474.0061	2552	128879.8537
2491	88138.51255	2522	225856.0804	2553	80683.12856
2492	198086.5012	2523	121377.0244	2554	108107.4271
2493	146285.1932	2524	132760.4029	2555	112896.3748
2494	167133.1415	2525	142351.2823	2556	98962.09073
2495	87930.16943	2526	140252.7705	2557	111754.6636
2496	249435.0501	2527	106207.2282	2558	141874.7977
2497	152738.2052	2528	120631.2754	2559	133453.7078
2498	103103.5463	2529	135095.2566	2560	150356.1745
2499	88436.16384	2530	114069.8768	2561	140598.8662
2500	127640.7875	2531	256238.6293	2562	128498.2009
2501	147150.414	2532	224923.5208	2563	146783.7961
2502	162122.5077	2533	193508.1154	2564	185356.9113
2503	93512.79532	2534	239508.6904	2565	121337.6744
2504	179136.5075	2535	290591.7567	2566	158818.4603
2505	221571.6817	2536	228242.0879	2567	122541.9295
2506	263301.1338	2537	232313.7565	2568	218925.5988
2507	298486.4622	2538	181071.1151	2569	212034.7003
2508	263496.5884	2539	194602.995	2570	125085.8674
2509	217205.5588	2540	172645.8499	2571	163847.0471
2510	220277.41	2541	174445.9823	2572	140448.958
2511	182873.3861	2542	157328.7968	2573	246168.1
2512	214128.9166	2543	124071.8064	2574	228283.3371

2575	128143.1682	2606	145318.5837	2637	170887.7023
2576	131504.0809	2607	211219.533	2638	246188.5728
2577	159007.339	2608	192405.5023	2639	189987.3218
2578	76291.75799	2609	167336.7045	2640	154543.9703
2579	72696.84065	2610	122000.4163	2641	110359.6551
2580	124074.8121	2611	143194.579	2642	189418.8365
2581	130968.6198	2612	153691.7089	2643	94095.68085
2582	121138.7736	2613	129164.5165	2644	129318.1234
2583	283270.6077	2614	129716.7599	2645	97372.59414
2584	174373.7453	2615	161560.5819	2646	96303.93626
2585	168991.4774	2616	134134.8107	2647	97345.68468
2586	213689.3838	2617	166464.7623	2648	138231.6744
2587	187384.8448	2618	191888.1129	2649	142514.1693
2588	157538.4665	2619	211703.2267	2650	146448.1005
2589	148109.7314	2620	188555.6075	2651	151997.0011
2590	224045.7728	2621	180686.1665	2652	453080.7478
2591	215374.84	2622	179306.9087	2653	254221.0874
2592	220815.792	2623	253213.0894	2654	245217.2116
2593	268003.05	2624	292037.9388	2655	380264.2724
2594	177825.6591	2625	284716.1485	2656	306624.075
2595	192326.0987	2626	169644.3224	2657	320945.9385
2596	332241.5724	2627	165983.5042	2658	314219.0335
2597	168912.0744	2628	533371.4666	2659	295943.7131
2598	322258.0721	2629	485179.6879	2660	345749.6811
2599	286252.7713	2630	337004.0312	2661	318481.0628
2600	118283.4232	2631	483665.8403	2662	340290.369
2601	142032.7388	2632	502627.1671	2663	267187.8722
2602	79384.21626	2633	317363.3592	2664	262413.7917
2603	94027.86894	2634	367151.0148	2665	320728.2063
2604	86897.4867	2635	149418.8391	2666	290269.8559
2605	73385.76481	2636	171982.1924	2667	167035.1604

2668	167421.0585	2699	172981.3872	2730	146845.3703
2669	174229.3415	2700	162640.4492	2731	121973.9671
2670	277304.7629	2701	152132.124	2732	126750.5274
2671	175160.1372	2702	113883.2523	2733	163611.6494
2672	197329.6438	2703	122906.8727	2734	147685.9242
2673	211221.087	2704	139633.3514	2735	131928.4891
2674	195702.0865	2705	119887.2096	2736	148592.8041
2675	158632.0795	2706	122263.3774	2737	127400.219
2676	186131.6889	2707	131035.8427	2738	126898.5805
2677	198999.6705	2708	134336.2695	2739	169878.0855
2678	259659.7449	2709	119506.9183	2740	125000.6973
2679	259766.1867	2710	125626.3607	2741	149842.0593
2680	261172.3778	2711	299128.5373	2742	159023.5568
2681	427586.605	2712	371782.1192	2743	153303.6599
2682	338297.793	2713	168343.3186	2744	156158.9279
2683	522303.7286	2714	148446.4843	2745	138055.1162
2684	312588.9597	2715	174304.7269	2746	135760.6949
2685	344674.8255	2716	150985.3062	2747	138301.4996
2686	273992.3489	2717	166107.0238	2748	120898.6311
2687	355560.7948	2718	222345.4521	2749	128415.4297
2688	229571.1598	2719	168584.5817	2750	132584.9794
2689	196947.8933	2720	179279.0761	2751	130082.9582
2690	450368.0048	2721	139987.0153	2752	215279.0852
2691	189217.0631	2722	169525.2111	2753	153172.2132
2692	136364.5162	2723	131081.2046	2754	195797.3817
2693	207563.8998	2724	128863.7193	2755	136148.3437
2694	132444.8319	2725	132905.1055	2756	83035.02343
2695	194872.9392	2726	148715.9367	2757	57248.82488
2696	184920.8124	2727	172641.4592	2758	85467.37023
2697	186748.1381	2728	173720.871	2759	151310.7039
2698	182665.5928	2729	155615.3414	2760	128017.4548

2761	145705.5214	2792	62957.1734	2823	258375.7777
2762	158446.6333	2793	194167.4181	2824	187079.7104
2763	183216.9955	2794	99470.76095	2825	150189.6681
2764	158748.7531	2795	126563.0116	2826	133030.4311
2765	285396.6989	2796	93291.4166	2827	138578.367
2766	142781.2501	2797	170901.3493	2828	218095.7807
2767	90621.30464	2798	118303.4467	2829	213363.0038
2768	124992.8559	2799	107401.322	2830	220730.4086
2769	126425.2232	2800	65028.76786	2831	182210.23
2770	144265.0864	2801	107379.1205	2832	227887.0486
2771	113678.4555	2802	132380.8104	2833	287449.3046
2772	115290.3275	2803	160393.1883	2834	217751.2596
2773	145989.7455	2804	125293.5289	2835	213642.2278
2774	137323.4255	2805	95804.91176	2836	192581.8633
2775	123930.0792	2806	80994.09626	2837	158130.8849
2776	141874.158	2807	156339.2972	2838	148591.5201
2777	144610.9078	2808	153009.5465	2839	176960.6962
2778	128810.7403	2809	136485.5932	2840	203811.0351
2779	119824.0955	2810	131874.9458	2841	210366.004
2780	107023.2151	2811	164423.4641	2842	225384.4828
2781	89825.76398	2812	144554.6334	2843	159501.6805
2782	103047.1038	2813	134939.0393	2844	176856.21
2783	95777.09711	2814	137791.2082	2845	123278.1327
2784	133013.9205	2815	100837.8722	2846	213469.3107
2785	131151.3817	2816	285883.5845	2847	202418.3529
2786	72412.85263	2817	150661.8245	2848	206922.3589
2787	115567.1056	2818	131223.2721	2849	204920.2562
2788	74914.13988	2819	164365.423	2850	286479.1277
2789	198361.8325	2820	139753.1642	2851	241157.2207
2790	87114.49487	2821	120383.0343	2852	228266.9767
2791	122003.0617	2822	209646.8311	2853	240997.7781

2854	137975.3468	2885	201339.92	2915	76336.21326
2855	211123.4407	2886	215173.9548	2916	83815.50536
2856	199314.6472	2887	83146.53542	2917	165428.4894
2857	190337.0328	2888	139315.6726	2918	110235.0722
2858	210955.7801	2889	59269.8345	2919	211095.869
2859	119980.4221	2890	80647.78411		
2860	122147.508	2891	123303.0431		
2861	122906.2072	2892	55453.69847		
2862	191986.1605	2893	101141.1587		
2863	127844.1295	2894	59011.67241		
2864	233895.8961	2895	354457.9186		
2865	141862.63	2896	287661.7934		
2866	131329.6374	2897	204242.7077		
2867	99685.27509	2898	136634.5948		
2868	103097.2128	2899	204847.8991		
2869	113570.9862	2900	165028.2528		
2870	127244.6568	2901	182885.5874		
2871	84778.11993	2902	203112.4458		
2872	49206.99424	2903	381510.8566		
2873	98888.44041	2904	360649.7599		
2874	142952.2689	2905	96031.35915		
2875	120568.7048	2906	200584.333		
2876	169362.8534	2907	112729.7742		
2877	138753.3482	2908	129632.6422		
2878	160303.1386	2909	138979.3396		
2879	139436.6398	2910	80404.35274		
2880	99645.06151	2911	77978.06359		
2881	161164.2005	2912	149909.9655		
2882	177924.3907	2913	90132.31864		
2883	172407.9666	2914	74486.13374		
2884	186541.8648				

Kaggle Competition Score, which fetched me 2100 rank in first try.

Competitions

Datasets

Notebooks

Discussion

Courses


...

Signed in as Ishan Yash

My Profile

My Account

Sign Out



House Prices: Advanced Regression Techniques

Predict sales prices and practice feature engineering, RFs, and gradient boosting

4,444 teams · Ongoing

Overview

Data

Notebooks

Discussion

Leaderboard

Rules

Team

My Submissions

Submit Predictions

Your most recent submission

Name	Submitted	Wait time	Execution time	Score
aiwp_task3.csv	6 minutes ago	0 seconds	0 seconds	0.13365

Complete

[Jump to your position on the leaderboard](#)

You may select up to 5 submissions to be used to count towards your final leaderboard score. If 5 submissions are not selected, they will be automatically chosen based on your best submission scores on the public leaderboard. In the event that automatic selection is not suitable, manual selection instructions will be provided in the competition rules or by official forum announcement.

Your final score may not be based on the same exact subset of data as the public leaderboard, but rather a different private data subset of your full submission — your public score is only a rough indication of what your final score is.