

```
In [1]: import numpy as np
import gzip
import urllib.request
import tarfile
import random
import string
from copy import deepcopy
import pandas as pd
from collections import defaultdict
import json
from sklearn.metrics import mean_squared_error
from sklearn import linear_model
from sklearn.svm import LinearSVC
import matplotlib.pyplot as plt
import re
import matplotlib as mpl
import matplotlib.pyplot as plt
import patsy
import statsmodels.api as sm
import seaborn as sns
%matplotlib notebook
```

```
In [2]: df_wine = pd.read_json("winemag-data-130k-v2.json")
```

```
In [3]: df_wine.columns.values
```

```
Out[3]: array(['country', 'description', 'designation', 'points', 'price',
              'province', 'region_1', 'region_2', 'taster_name',
              'taster_twitter_handle', 'title', 'variety', 'winery'],
             dtype=object)
```

## Dropping the none

```
In [4]: df_wine = df_wine[df_wine.country != 'nan']
df_wine = df_wine[df_wine.description != 'nan']
df_wine = df_wine[df_wine.points != 'nan']
df_wine = df_wine[df_wine.price != 'nan']
df_wine = df_wine[df_wine.title != 'nan']
df_wine = df_wine[df_wine.variety != 'nan']
df_wine = df_wine[df_wine.winery != 'nan']
```

```
C:\Users\d3v1n\Anaconda3\lib\site-packages\pandas\core\ops.py:1649: FutureWarning: elementwise comparison failed; returning scalar instead, but in the future will perform elementwise comparison
    result = method(y)
```

## Get the description of the data

In [5]: `df_wine.head()`

Out[5]:

	country	description	designation	points	price	province	region_1	region_2	taster_name
0	Italy	Aromas include tropical fruit, broom, brimston...	Vulkà Bianco	87	NaN	Sicily & Sardinia	Etna	None	Ker O'Ke
1	Portugal	This is ripe and fruity, a wine that is smooth...	Avidagos	87	15.0	Douro	None	None	Roger Vos
2	US	Tart and snappy, the flavors of lime flesh and...	None	87	14.0	Oregon	Willamette Valley	Willamette Valley	Paul Gregu
3	US	Pineapple rind, lemon pith and orange blossom ...	Reserve Late Harvest	87	13.0	Michigan	Lake Michigan Shore	None	Alexandi Peartre
4	US	Much like the regular bottling from 2012, this...	Vintner's Reserve Wild Child Block	87	65.0	Oregon	Willamette Valley	Willamette Valley	Paul Gregu

In [6]: `df_wine['points'].mean()`

Out[6]: 88.44713820775404

In [7]: `df_wine['taster_name'].value_counts().count()`

Out[7]: 19

In [8]: `df_wine['title'].value_counts().count()`

Out[8]: 118840

```
In [9]: df_wine['winery'].value_counts().count()
```

```
Out[9]: 16757
```

```
In [10]: df_wine['country'].value_counts().count()
```

```
Out[10]: 43
```

## Get the continent

```
In [11]: def continent(country):  
    if country in ['US', 'Canada', 'Mexico']:  
        return 'North America'  
    elif country in ['France', 'Italy', 'Spain', 'Portugal', 'Austria', 'Ge  
        , 'Turkey', 'Slovenia', 'Georgia', 'England', 'Croatia  
        , 'Serbia', 'Cyprus', 'Switzerland', 'Luxembourg', 'Bo  
        return 'Europe'  
    elif country in ['Chile', 'Argentina', 'Uruguay', 'Brazil', 'Peru']:  
        return 'South America'  
    elif country in ['Israel', 'India', 'China']:  
        return 'Asia'  
    elif country in ['Australia', 'New Zealand']:  
        return 'Oceania'  
    elif country in ['South Africa', 'Egypt', 'Morocco']:  
        return 'Africa'  
    else:  
        return np.nan
```

```
In [12]: df_wine['continent'] = df_wine['country'].apply(continent)
```

## Get the year and age from the title of the wine

```
In [13]: ▶ def getYear(string):
    splitStr = []
    splitStr = string.split(' ')
    for a in splitStr:
        if a.isnumeric() == True:
            if int(a) >= 1934 and int(a) <= 2020:
                return a
            else:
                continue
        else:
            continue
    return np.nan
```

```
In [14]: ▶ df_wine['year'] = df_wine['title'].apply(getYear)
```

```
In [15]: ▶ def getAge(year):
    if type(year) == float:
        return np.nan
    else:
        age = 2018 - int(year)
        return age
```

```
In [16]: ▶ df_wine['age'] = df_wine['year'].apply(getAge)
```

```
In [17]: ▶ df_wine
```

129943	Italy	A blend of Nero d'Avola and Syrah, this convey...	Adènzia	90	29.0	Sicily & Sardinia	Sicilia
129944	Israel	Deep garnet in the glass, this has a nose of b...	Special Reserve Winemakers' Choice	90	25.0	Galilee	None
129945	US	Hailing from one of the more popular vineyards...	Jurassic Park Vineyard Old Vines	90	20.0	California	Santa Ynez Valley
129946	Germany	Plump, clingy peach and honey	Dom	90	17.0	Mosel	None

## Get the average rating that the taster give to know how generous they are

```
In [18]: taster = []
taster = df_wine['taster_name'].value_counts().keys().to_list()
average_taster = []
for a in taster:
    average_taster.append(df_wine[df_wine['taster_name'] == a].points.mean())
df_taster = pd.DataFrame({'taster_name':taster, 'points':average_taster})
df_taster = df_taster.sort_values(by=['points'], ascending=False).reset_index()
df_taster = df_taster.set_index('taster_name')
```

```
In [19]: df_taster
```

Out[19]:

	points
taster_name	
Anne Krebiehl MW	90.562551
Matt Kettmann	90.008686
Virginie Boone	89.213379
Mike DeSimone	89.101167
Paul Gregutt	89.082564
Kerin O'Keefe	88.867947
Sean P. Sullivan	88.755739
Roger Voss	88.708003
Jim Gordon	88.626287
Joe Czerwinski	88.536235
Anna Lee C. Iijima	88.415629
Jeff Jenssen	88.319756
Christina Pickard	87.833333
Lauren Buzzeo	87.739510
Michael Schachner	86.907493
Fiona Adams	86.888889
Susan Kostrzewa	86.609217
Carrie Dykes	86.395683
Alexander Peartree	85.855422

## Get the continent that produces the best wines

```
In [20]: ▶ continent = []
continent = df_wine['continent'].value_counts().keys().to_list()
average_continent = []
for a in continent:
    average_continent.append(df_wine[df_wine['continent'] == a].points.mean())
df_continent = pd.DataFrame({'continent':continent, 'points':average_continent})
df_continent = df_continent.sort_values(by=['points'], ascending=False).reset_index()
df_continent = df_continent.set_index('continent')
```

```
In [21]: ▶ df_continent
```

Out[21]:

	points
continent	
Europe	88.608982
North America	88.563276
Asia	88.502913
Oceania	88.475454
Africa	88.063636
South America	86.577583

## Get the country that produces the best wines

```
In [22]: ▶ country = []
country = df_wine['country'].value_counts().keys().to_list()
average_country = []
for a in country:
    average_country.append(df_wine[df_wine['country'] == a].points.mean())
df_country = pd.DataFrame({'country':country, 'points':average_country})
df_country = df_country.sort_values(by=['points'], ascending=False).reset_index()
df_country = df_country.set_index('country')
```

In [23]: `df_country`

Out[23]:

	points
country	
England	91.581081
India	90.222222
Austria	90.101345
Germany	89.851732
Canada	89.369650
Hungary	89.191781
China	89.000000
France	88.845109
Luxembourg	88.666667
Australia	88.580507
Switzerland	88.571429
Morocco	88.571429
US	88.563720
Italy	88.562231
Israel	88.471287
New Zealand	88.303030
Portugal	88.250220
Turkey	88.088889
Slovenia	88.068966
South Africa	88.056388
Bulgaria	87.936170
Georgia	87.686047
Lebanon	87.685714
Serbia	87.500000
Armenia	87.500000
Spain	87.288337
Greece	87.283262
Czech Republic	87.250000
Croatia	87.219178
Moldova	87.203390
Cyprus	87.181818
Slovakia	87.000000

	points
country	
Macedonia	86.833333
Uruguay	86.752294
Argentina	86.710263
Bosnia and Herzegovina	86.500000
Chile	86.493515
Romania	86.400000
Mexico	85.257143
Brazil	84.673077
Ukraine	84.071429
Egypt	84.000000
Peru	83.562500

## Get the winery that produces the best wines

```
In [24]: ► winery = df_wine['winery'].unique()  
winery
```

```
Out[24]: array(['Nicosia', 'Quinta dos Avidagos', 'Rainstorm', ...,  
               'Mas de Pampelonne', 'Bodegas Eidosela', 'Penedo Borges'],  
              dtype=object)
```

```
In [25]: ► avg1 = []  
for i in winery:  
    avg1.append(df_wine[df_wine.winery == i]['points'].mean())
```



```
In [26]:  wavg_points = {'Winery':winery, 'Average Points': avg1}

          wavg_points = pd.DataFrame(wavg_points)

          wavg_points = wavg_points.sort_values(by=['Average Points'], ascending=False)
          wavg_points
```

Out[26]:

	Winery	Average Points
11466	Araujo	98.000000
16486	Gandona	97.000000
16485	Ovid	97.000000
15600	J.L. Chave	97.000000
7914	Salon	96.800000
1300	Tenuta dell'Ornellaia	96.700000
7173	Château Pétrus	96.666667
11901	Cardinale	96.000000
14239	Colgin	96.000000
7353	Hors Categorie	96.000000
13853	Lewis	96.000000
14634	Patrimony	96.000000
16272	D'Alfonso-Curran	96.000000
16487	Bryant Family	96.000000
12160	Domaine des Lambrays	96.000000
16573	Semper	96.000000
15988	Celani Family Vineyards	96.000000
14633	Barons de Rothschild	96.000000
11169	Château Climens	95.750000
4845	Château d'Yquem	95.666667
5101	Château Léoville Barton	95.666667
15728	Au Sommet	95.500000
5861	Dolce	95.500000
15960	Masseto	95.500000
14635	Crown Point	95.500000
11784	Vieux Château Certan	95.400000
16307	Arkenstone	95.333333
3149	Wayfarer	95.333333
9216	Yao Ming	95.333333
4673	Mascarello Giuseppe e Figlio	95.200000

	Winery	Average Points
...	...	...
16266	Soleau	80.000000
14301	Männle	80.000000
15995	Rai	80.000000
16003	Campo Marin	80.000000
14417	Ronan	80.000000
14632	TorresManur	80.000000
11592	Nine x Nine	80.000000
10509	Vega Murillo	80.000000
13195	Corinto	80.000000
16271	T	80.000000
16304	Torito Bravo	80.000000
16619	Toca Diamonte	80.000000
15089	Bodegas del Mundo	80.000000
16484	Rojo Mojo	80.000000
15088	Immersion	80.000000
15086	Château Majoureau	80.000000
15613	Catharine Valley	80.000000
9954	Tenimenti Montagnana	80.000000
15615	Woomera	80.000000
16722	BookCliff	80.000000
16723	Reyter	80.000000
15084	Vega de Tera	80.000000
15083	Hawthorne Ridge	80.000000
16620	Unión Vinícola del Este	80.000000
14630	Aromal	80.000000
16310	Gran Familia	80.000000
7130	Villa Viña	80.000000
16334	Capay Valley	80.000000
14868	Moss Roxx	80.000000
8075	California's Jewel	80.000000

16757 rows × 2 columns

## Find the variety that the tasters like

```
In [27]: ► grapes = df_wine['variety'].unique()
grapes

    Touriga Nacional ,  Carmenere ,  Albarino ,  Petit Manseng ,
    'Rosé', 'Zinfandel', 'Vernaccia', 'Rosato', 'Grüner Veltliner',
    'Viognier', 'Vermentino', 'Grenache Blanc', 'Syrah', 'Nebbiol
o',
    'Shiraz-Cabernet Sauvignon', 'Pinot Blanc', 'Alsace white blen
d',
    'Barbera', 'Rhône-style Red Blend', 'Portuguese White', 'Gracia
no',
    'Tannat-Cabernet', 'Sauvignon', 'Sangiovese Grosso', 'Torronté
s',
    'Prugnolo Gentile', 'G-S-M', 'Verdejo', 'Fumé Blanc', 'Furmin
t',
    'Pinot Bianco', 'Bonarda', 'Shiraz', 'Montepulciano', 'Moscat
o',
    'Grenache', 'Ugni Blanc-Colombard', 'Syrah-Viognier',
    'Blaufränkisch', 'Friulano', 'Assyrtico', 'Carignan-Grenache',
    'Sagrantino', 'Savagnin', 'Cabernet Sauvignon-Syrah', 'Prosecc
o',
    'Vignoles', 'Sparkling Blend', 'Muscat', 'Muscadelle',
    'Shiraz-Viognier', 'Garganega', 'Pinot Grigio', 'Tempranillo',
```

```
In [28]: ► avg = []
for i in grapes:
    avg.append(df_wine[df_wine.variety == i]['points'].mean())
```

```
In [29]:  ▸ gavg_points = {'Grapes':grapes, 'Average Points': avg}

gavg_points = pd.DataFrame(gavg_points)

gavg_points = gavg_points.sort_values(by=['Average Points'], ascending=False)
gavg_points
```

Out[29]:

	Grapes	Average Points
399	Tinta del Pais	95.000000
486	Terrantez	95.000000
633	Gelber Traminer	95.000000
487	Bual	94.142857
558	Riesling-Chardonnay	94.000000
661	Sercial	94.000000
678	Ramisco	93.000000
681	Garnacha-Cariñena	93.000000
500	Blauburgunder	93.000000
653	Sirica	92.500000
83	Muscadelle	92.500000
463	Rosenmuskateller	92.500000
603	Roviello	92.500000
688	Babosa Negro	92.000000
696	Torontel	92.000000
703	Pignolo	92.000000
674	Centesimino	92.000000
590	Chardonnay Weissburgunder	92.000000
704	Caprettone	92.000000
640	Sauvignon Blanc-Assyrtiko	92.000000
637	Tinta Negra Mole	92.000000
694	Moscato di Noto	92.000000
692	Kotsifali	92.000000
544	Vespaiolo	92.000000
466	Malbec-Cabernet	91.666667
684	Chardonnay-Albariño	91.500000
269	Syrah-Petit Verdot	91.333333
409	Loin de l'Oeil	91.333333
343	Sämling	91.200000
374	Picolit	91.111111

	<b>Grapes</b>	<b>Average Points</b>
...	...	...
217	Seyval Blanc	84.187500
490	Dofia Blanca	84.000000
698	Viognier-Valdiguie	84.000000
292	Norton	84.000000
655	Espadeiro	84.000000
350	Siegeerrebe	84.000000
539	Verdosilla	84.000000
229	Kinali Yapincak	84.000000
707	Bobal-Cabernet Sauvignon	84.000000
283	Asprinio	84.000000
423	Pigato	84.000000
417	Cabernet Sauvignon-Barbera	84.000000
562	Pinotage-Merlot	84.000000
355	Brachetto	83.800000
195	Chambourcin	83.722222
474	Tempranillo Blanco	83.666667
667	Premsal	83.000000
329	País	83.000000
530	Paralleda	83.000000
412	Forcallà	83.000000
627	Chardonel	83.000000
706	Athiri	83.000000
677	Merlot-Petite Verdot	83.000000
582	Gragnano	82.500000
548	Aidani	82.000000
671	Picapoll	82.000000
643	Shiraz-Tempranillo	82.000000
536	Airen	81.666667
666	Chancellor	80.500000
641	None	NaN

708 rows × 2 columns

**Get the best wines**

```
In [30]: wine = df_wine['title'].unique()  
wine
```

```
Out[30]: array(['Nicosia 2013 Vulkà Bianco (Etna)',  
               'Quinta dos Avidagos 2011 Avidagos Red (Douro)',  
               'Rainstorm 2013 Pinot Gris (Willamette Valley)', ...,  
               'Domaine Gresser 2013 Kritt Gewurztraminer (Alsace)',  
               'Domaine Marcel Deiss 2012 Pinot Gris (Alsace)',  
               'Domaine Schoffit 2012 Lieu-dit Harth Cuvée Caroline Gewurztraminer  
               (Alsace)'],  
          dtype=object)
```

```
In [31]: avg = []  
for i in wine:  
    avg.append(df_wine[df_wine.title == i]['points'].mean())
```

```
In [32]: wine_avg_points = {'Wine':wine, 'Average Points': avg}

wine_avg_points = pd.DataFrame(wine_avg_points)

wine_avg_points = wine_avg_points.sort_values(by=['Average Points'], ascend
wine_avg_points[wine_avg_points['Average Points'] == 100]
```

Out[32]:

	Wine	Average Points
84249	Salon 2006 Le Mesnil Blanc de Blancs Brut Char...	100.0
103443	Château Léoville Las Cases 2010 Saint-Julien	100.0
41019	Casa Ferreirinha 2008 Barca-Velha Red (Douro)	100.0
56012	Château Léoville Barton 2010 Saint-Julien	100.0
113445	Cayuse 2008 Bionic Frog Syrah (Walla Walla Val...	100.0
44375	Cardinale 2006 Cabernet Sauvignon (Napa Valley)	100.0
105287	Charles Smith 2006 Royal City Syrah (Columbia ...	100.0
44358	Biondi Santi 2010 Riserva (Brunello di Montal...	100.0
103440	Château Lafite Rothschild 2010 Pauillac	100.0
103441	Casanova di Neri 2007 Cerretalto (Brunello di...	100.0
38250	Tenuta dell'Ornellaia 2007 Masseto Merlot (Tos...	100.0
103442	Château Cheval Blanc 2010 Saint-Émilion	100.0
7295	Avignonesi 1995 Occhio di Pernice (Vin Santo ...	100.0
108791	Verité 2007 La Muse Red (Sonoma County)	100.0
106189	Quinta do Noval 2011 Nacional Vintage (Port)	100.0
35648	Krug 2002 Brut (Champagne)	100.0
84248	Louis Roederer 2008 Cristal Vintage Brut (Cha...	100.0

## Find out if the quality of the wine increases by age

```
In [33]: age = []
age = df_wine['age'].value_counts().keys().to_list()
average_age = []
for a in age:
    average_age.append(df_wine[df_wine['age'] == a].points.mean())
df_age = pd.DataFrame({'age':age, 'points':average_age})
df_age = df_age.sort_values(by=['age'], ascending=True).reset_index(drop=True)
df_age = df_age.set_index('age')
```

In [34]: `df_age`

Out[34]:

	points
age	
1.0	85.545455
2.0	87.687077
3.0	88.519200
4.0	88.867283
5.0	88.952189
6.0	88.774052
7.0	88.277672
8.0	88.262020
9.0	88.334987
10.0	88.203791
11.0	88.163051
12.0	88.216216
13.0	88.351785
14.0	88.761556
15.0	88.404682
16.0	87.814208
17.0	87.392185
18.0	87.402375
19.0	87.616225
20.0	87.500907
21.0	89.115512
22.0	88.462687
23.0	89.400000
24.0	88.434783
25.0	87.666667
26.0	90.812500
27.0	88.500000
28.0	89.800000
29.0	88.666667
30.0	89.600000
31.0	92.000000
32.0	91.000000



	points
age	
33.0	90.750000
34.0	94.500000
35.0	90.500000
36.0	95.000000
38.0	94.000000
40.0	88.500000
42.0	88.000000
44.0	93.000000
45.0	96.000000
49.0	98.000000
50.0	95.000000
51.0	94.000000
52.0	95.000000
53.0	94.000000
54.0	93.000000
55.0	92.666667
57.0	95.000000
61.0	95.000000
66.0	95.500000
71.0	93.000000
73.0	95.000000
77.0	93.000000
83.0	94.000000
84.0	93.000000

**Find out if the prices of the wine matters**

```
In [35]: ▶ price = []
price = df_wine['price'].value_counts().keys().to_list()
average_price = []
for a in price:
    average_price.append(df_wine[df_wine['price'] == a].points.mean())
df_price = pd.DataFrame({'price':price, 'points':average_price})
df_price = df_price.sort_values(by=['price'], ascending=True).reset_index(d
df_price = df_price.set_index('price')
```

In [36]: `df_price`

Out[36]:

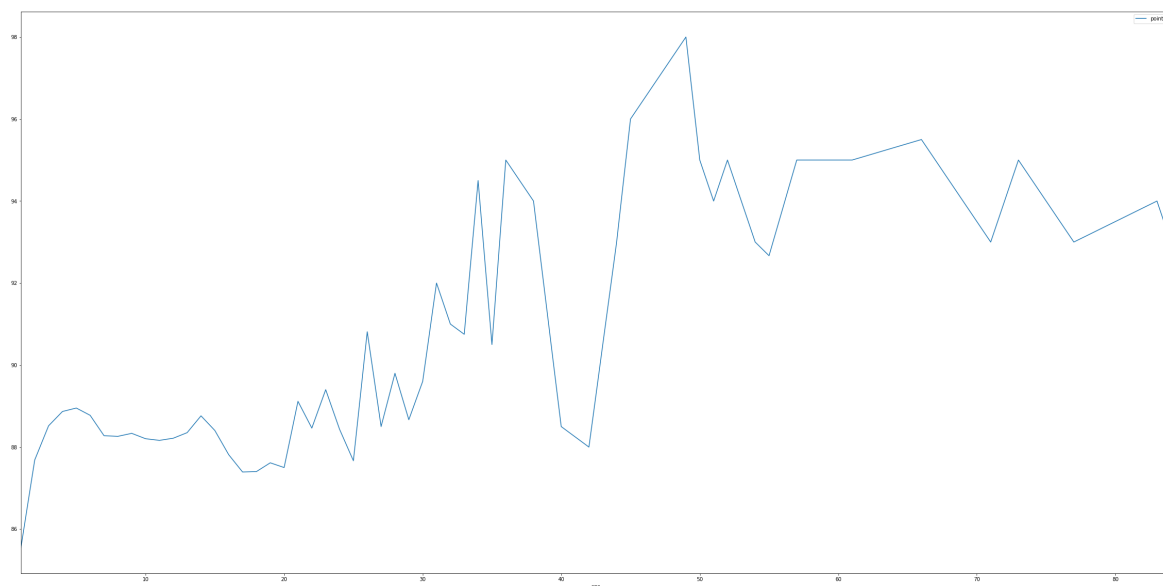
	points
price	
4.0	84.272727
5.0	83.586957
6.0	84.341667
7.0	84.450346
8.0	84.628924
9.0	85.036594
10.0	85.239314
11.0	85.471331
12.0	85.807067
13.0	86.054663
14.0	86.474650
15.0	86.654962
16.0	87.011841
17.0	87.156567
18.0	87.249642
19.0	87.588778
20.0	87.620893
21.0	87.809253
22.0	87.914507
23.0	87.945773
24.0	88.021231
25.0	88.210164
26.0	88.226260
27.0	88.449288
28.0	88.388171
29.0	88.608508
30.0	88.667340
31.0	89.380435
32.0	88.851248
33.0	88.983533
...	...
698.0	97.000000

	points
price	
710.0	95.000000
750.0	92.000000
757.0	98.000000
764.0	94.000000
767.0	96.000000
770.0	94.500000
775.0	96.250000
780.0	91.000000
790.0	87.000000
800.0	95.000000
820.0	96.000000
848.0	100.000000
850.0	97.333333
886.0	97.000000
900.0	94.000000
932.0	97.000000
973.0	95.000000
980.0	94.000000
1000.0	95.500000
1100.0	95.500000
1125.0	94.000000
1200.0	96.000000
1300.0	96.000000
1500.0	100.000000
1900.0	98.000000
2000.0	96.500000
2013.0	91.000000
2500.0	96.000000
3300.0	88.000000

390 rows × 1 columns

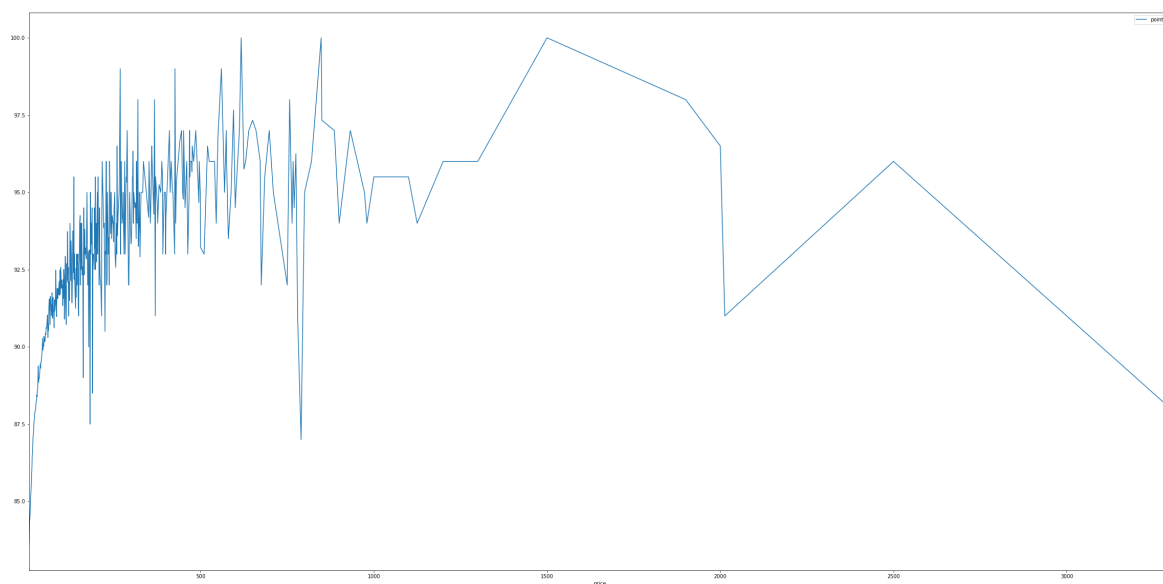
```
In [37]: df_age.plot(kind='line', figsize=(40,20))
```

Out[37]: <matplotlib.axes.\_subplots.AxesSubplot at 0x1cfb9897048>



```
In [38]: df_price.plot(kind='line', figsize=(40,20))
```

Out[38]: <matplotlib.axes.\_subplots.AxesSubplot at 0x1cfb98c7588>



In [39]: `df_taster`

Out[39]:

	points
taster_name	
Anne Krebiehl MW	90.562551
Matt Kettmann	90.008686
Virginie Boone	89.213379
Mike DeSimone	89.101167
Paul Gregutt	89.082564
Kerin O'Keefe	88.867947
Sean P. Sullivan	88.755739
Roger Voss	88.708003
Jim Gordon	88.626287
Joe Czerwinski	88.536235
Anna Lee C. Iijima	88.415629
Jeff Jenssen	88.319756
Christina Pickard	87.833333
Lauren Buzzeo	87.739510
Michael Schachner	86.907493
Fiona Adams	86.888889
Susan Kostrzewa	86.609217
Carrie Dykes	86.395683
Alexander Peartree	85.855422

**Give the suggestion to each taster based on their preferred winery and variety**

```
In [40]: ▶ def getSuggestion(dfWinery, dfVariety, tasterName):
        return df_wine.loc[((df_wine['winery'] == dfWinery.index.tolist()[0])
                             | (df_wine['winery'] == dfWinery.index.to
                             | (df_wine['winery'] == dfWinery.index.to
                             | (df_wine['winery'] == dfWinery.index.to
                             | (df_wine['winery'] == dfWinery.index.to
                             | (df_wine['variety'] == dfVariety.index.
                             | (df_wine['variety'] == dfVariety.index.
                             | (df_wine['variety'] == dfVariety.index.
                             | (df_wine['variety'] == dfVariety.index.
                             & (df_wine['taster_name'] != tasterName))]
```

```
In [41]: ▶ def getSuggestionTwitter(dfWinery, dfVariety, tasterName):
        return df_wine.loc[((df_wine['winery'] == dfWinery.index.tolist()[0])
                             | (df_wine['winery'] == dfWinery.index.to
                             | (df_wine['winery'] == dfWinery.index.to
                             | (df_wine['winery'] == dfWinery.index.to
                             | (df_wine['winery'] == dfWinery.index.to
                             | (df_wine['variety'] == dfVariety.index.
                             | (df_wine['variety'] == dfVariety.index.
                             | (df_wine['variety'] == dfVariety.index.
                             | (df_wine['variety'] == dfVariety.index.
                             & (df_wine['taster_twitter_handle'] != tas
```

```
In [42]: ▶ df_voss = df_wine.loc[df_wine['taster_name'] == 'Roger Voss']
        df_voss = df_voss.reset_index(drop=True)
```

```
In [43]: ▶ winery_voss = []
        winery_voss = df_voss['winery'].value_counts().keys().to_list()
        average_winery_voss = []
        for a in winery_voss:
            average_winery_voss.append(df_voss[df_voss['winery'] == a].points.mean()
        df_winery_voss = pd.DataFrame({'winery': winery_voss, 'points': average_winer
        df_winery_voss = df_winery_voss.sort_values(by=['points'], ascending=False)
        df_winery_voss = df_winery_voss.set_index('winery')
```

```
In [44]: ▶ variety_voss = []
        variety_voss = df_voss['variety'].value_counts().keys().to_list()
        average_variety_voss = []
        for a in variety_voss:
            average_variety_voss.append(df_voss[df_voss['variety'] == a].points.mea
        df_variety_voss = pd.DataFrame({'variety': variety_voss, 'points': average_va
        df_variety_voss = df_variety_voss.sort_values(by=['points'], ascending=Fals
        df_variety_voss = df_variety_voss.set_index('variety')
```

```
In [45]: df_suggestion_voss = getSuggestion(df_winery_voss, df_variety_voss, 'Roger')
df_suggestion_voss = df_suggestion_voss.sort_values(by=['points'], ascending=False)
df_suggestion_voss
```

Out[45]:

	country	description	designation	points	price	province	region_1	region_2	ta
75556	France	An enormously complex and rich wine that almos...	Le Mesnil Blanc de Blancs Brut	96	216.0	Champagne	Champagne	None	
18792	France	Still seems quite young —or maybe it will always...	Le Mesnil Blanc de Blancs Brut	93	200.0	Champagne	Champagne	None	
65796	Germany	Slight hint of sulfur at first, but it blows o...	Trocken	90	25.0	Pfalz	None	None	
67382	France	A rich nose of honey, marmalade and yeast open...	Brut	88	230.0	Champagne	Champagne	None	
67389	France	Krug prefers to call this wine multivintage ra...	Grande Cuvée Brut	88	160.0	Champagne	Champagne	None	

```
In [46]: df_schachner = df_wine.loc[df_wine['taster_name'] == 'Michael Schachner']
df_schachner = df_schachner.reset_index(drop=True)
```

```
In [47]: winery_schachner = []
winery_schachner = df_schachner['winery'].value_counts().keys().to_list()
average_winery_schachner = []
for a in winery_schachner:
    average_winery_schachner.append(df_schachner[df_schachner['winery'] == a])
df_winery_schachner = pd.DataFrame({'winery':winery_schachner, 'points':average_winery_schachner})
df_winery_schachner = df_winery_schachner.sort_values(by=['points'], ascending=False)
df_winery_schachner = df_winery_schachner.set_index('winery')
```



```
In [48]: ► variety_schachner = []
          variety_schachner = df_schachner['variety'].value_counts().keys().to_list()
          average_variety_schachner = []
          for a in variety_schachner:
              average_variety_schachner.append(df_schachner[df_schachner['variety'] =
df_variety_schachner = pd.DataFrame({'variety':variety_schachner, 'points':
df_variety_schachner = df_variety_schachner.sort_values(by=['points'], asce
df_variety_schachner = df_variety_schachner.set_index('variety')
```

```
In [49]: ► df_suggestion_schachner = getSuggestion(df_winery_schachner, df_variety_sch
df_suggestion_schachner = df_suggestion_schachner.sort_values(by=['points'])
df_suggestion_schachner
```

Out[49]:

	country	description	designation	points	price	province	region_1	region_2	taster_1
39286	Italy	A perfect wine from a classic vintage, the 200...	Masseto	100	460.0	Tuscany	Toscana	None	
79104	Italy	Even better than the highly acclaimed 2001 vin...	Masseto	99	250.0	Tuscany	Toscana	None	
79105	Italy	This is a hopelessly gorgeous wine with so muc...	Ornellaia	98	175.0	Tuscany	Bolgheri Superiore	None	
45801	Italy	Consistently among Italy's top-scoring wines, ...	Ornellaia	97	200.0	Tuscany	Bolgheri Superiore	None	
39291	Italy	This wine shows growing intensity the longer i...	Masseto	97	460.0	Tuscany	Toscana	None	

```
In [50]: ► df_keefe = df_wine.loc[df_wine['taster_name'] == 'Kerin O'Keefe']
          df_keefe = df_keefe.reset_index(drop=True)
```

```
In [51]: ► winery_keefe = []
winery_keefe = df_keefe['winery'].value_counts().keys().to_list()
average_winery_keefe = []
for a in winery_keefe:
    average_winery_keefe.append(df_keefe[df_keefe['winery'] == a].points.mean())
df_winery_keefe = pd.DataFrame({'winery':winery_keefe, 'points':average_winery_keefe})
df_winery_keefe = df_winery_keefe.sort_values(by=['points'], ascending=False)
df_winery_keefe = df_winery_keefe.set_index('winery')
```

```
In [52]: ► variety_keefe = []
variety_keefe = df_keefe['variety'].value_counts().keys().to_list()
average_variety_keefe = []
for a in variety_keefe:
    average_variety_keefe.append(df_keefe[df_keefe['variety'] == a].points.mean())
df_variety_keefe = pd.DataFrame({'variety':variety_keefe, 'points':average_variety_keefe})
df_variety_keefe = df_variety_keefe.sort_values(by=['points'], ascending=False)
df_variety_keefe = df_variety_keefe.set_index('variety')
```

```
In [53]: df_suggestion_keefe = getSuggestion(df_winery_keefe, df_variety_keefe, 'Ker
df_suggestion_keefe = df_suggestion_keefe.sort_values(by=['points'], ascend
df_suggestion_keefe
```

Out[53]:

	country	description	designation	points	price	province	region_1	region_2	taster_
39293	Italy	This beautiful vintage of Paleo (an oak-aged e...	Paleo	96	125.0	Tuscany	Toscana	None	
79108	Italy	This Cabernet Franc is beautiful, with incredi...	Paleo	96	105.0	Tuscany	Toscana	None	
109418	Italy	Poggio al Tesoro, located in coastal Tuscany, ...	Dedicato a Walter	95	90.0	Tuscany	Toscana	None	
83370	US	A rare Cabernet Franc from Merry-vale, and a s...	None	95	90.0	California	Napa Valley	Napa	
74811	Italy	This 100% Cabernet Franc is dedicated to the l...	Dedicato a Walter	95	80.0	Tuscany	Toscana	None	

```
In [54]: df_boone = df_wine.loc[df_wine['taster_name'] == 'Virginie Boone']
df_boone = df_boone.reset_index(drop=True)
```

```
In [55]: ► winery_boone = []
winery_boone = df_boone['winery'].value_counts().keys().to_list()
average_winery_boone = []
for a in winery_boone:
    average_winery_boone.append(df_boone[df_boone['winery'] == a].points.mean())
df_winery_boone = pd.DataFrame({'winery': winery_boone, 'points': average_winery_boone})
df_winery_boone = df_winery_boone.sort_values(by='points', ascending=False)
df_winery_boone = df_winery_boone.set_index('winery')
```

```
In [56]: ► variety_boone = []
variety_boone = df_boone['variety'].value_counts().keys().to_list()
average_variety_boone = []
for a in variety_boone:
    average_variety_boone.append(df_boone[df_boone['variety'] == a].points.mean())
df_variety_boone = pd.DataFrame({'variety': variety_boone, 'points': average_variety_boone})
df_variety_boone = df_variety_boone.sort_values(by='points', ascending=False)
df_variety_boone = df_variety_boone.set_index('variety')
```

```
In [57]: df_suggestion_boone = getSuggestion(df_winery_boone, df_variety_boone, 'Vir
df_suggestion_boone = df_suggestion_boone.sort_values(by=['points'], ascend
df_suggestion_boone
```

Out[57]:

	country	description	designation	points	price	province	region_1	region_2	taster
103115	US	The tannins are so fierce on this baby Caberne...	None	95	65.0	California	Howell Mountain	Napa	
118069	US	The tannins are so fierce on this baby Caberne...	None	95	55.0	California	Howell Mountain	Napa	
14352	US	This is a big, powerful Bordeaux blend that cl...	Proprietary Red Wine	95	82.0	California	Napa Valley	Napa	
76584	Argentina	As far as Argentine Syrah goes, this vintage o...	Isca	94	75.0	Mendoza Province	Mendoza	None	! Sch
103120	US	Hard to describe the balance and elegance of t...	None	94	75.0	California	Howell Mountain	Napa	

```
In [58]: df_gregutt = df_wine.loc[df_wine['taster_name'] == 'Paul Gregutt']
df_gregutt = df_gregutt.reset_index(drop=True)
```

```
In [59]: ► winery_gregutt = []
winery_gregutt = df_gregutt['winery'].value_counts().keys().to_list()
average_winery_gregutt = []
for a in winery_gregutt:
    average_winery_gregutt.append(df_gregutt[df_gregutt['winery'] == a].poi
df_winery_gregutt = pd.DataFrame({'winery':winery_gregutt, 'points':average
df_winery_gregutt = df_winery_gregutt.sort_values(by=['points'], ascending=
df_winery_gregutt = df_winery_gregutt.set_index('winery')
```

```
In [60]: ► variety_gregutt = []
variety_gregutt = df_gregutt['variety'].value_counts().keys().to_list()
average_variety_gregutt = []
for a in variety_gregutt:
    average_variety_gregutt.append(df_gregutt[df_gregutt['variety'] == a].p
df_variety_gregutt = pd.DataFrame({'variety':variety_gregutt, 'points':aver
df_variety_gregutt = df_variety_gregutt.sort_values(by=['points'], ascendin
df_variety_gregutt = df_variety_gregutt.set_index('variety')
```

```
In [61]: ► df_suggestion_gregutt = getSuggestion(df_winery_gregutt, df_variety_gregutt)
df_suggestion_gregutt = df_suggestion_gregutt.sort_values(by=['points'], as
df_suggestion_gregutt
```

Out[61]:

	country	description	designation	points	price	province	region_1	region_2	tas
113929	US	In 2005 Charles Smith introduced three high-en...	Royal City	100	80.0	Washington	Columbia Valley (WA)	Columbia Valley	Pa
123545	US	Initially a rather subdued Frog; as if it has ...	Bionic Frog	100	80.0	Washington	Walla Walla Valley (WA)	Columbia Valley	Pa
47428	US	In a vintage that produced the finest overall ...	En Chamberlin Vineyard	99	75.0	Oregon	Walla Walla Valley (OR)	Oregon Other	Pa
1556	US	The flagship wine from Quilceda Creek offers e...	None	99	125.0	Washington	Columbia Valley (WA)	Columbia Valley	Pa
47894	US	Dark and concentrated, this sends up complex a...	En Chamberlin Vineyard	99	75.0	Oregon	Walla Walla Valley (OR)	Oregon Other	Pa

```
In [62]: ► df_kettmann = df_wine.loc[df_wine['taster_name'] == 'Matt Kettmann']
df_kettmann = df_kettmann.reset_index(drop=True)
```

```
In [63]: ► winery_kettmann = []
winery_kettmann = df_kettmann['winery'].value_counts().keys().to_list()
average_winery_kettmann = []
for a in winery_kettmann:
    average_winery_kettmann.append(df_kettmann[df_kettmann['winery'] == a].
df_winery_kettmann = pd.DataFrame({'winery':winery_kettmann, 'points':avera
df_winery_kettmann = df_winery_kettmann.sort_values(by=['points'], ascendin
df_winery_kettmann = df_winery_kettmann.set_index('winery')
```

```
In [64]: ► variety_kettmann = []
variety_kettmann = df_kettmann['variety'].value_counts().keys().to_list()
average_variety_kettmann = []
for a in variety_kettmann:
    average_variety_kettmann.append(df_kettmann[df_kettmann['variety'] == a
df_variety_kettmann = pd.DataFrame({'variety':variety_kettmann, 'points':av
df_variety_kettmann = df_variety_kettmann.sort_values(by=['points'], ascend
df_variety_kettmann = df_variety_kettmann.set_index('variety')
```



```
In [65]: df_suggestion_kettmann = getSuggestion(df_winery_kettmann, df_variety_kettm
df_suggestion_kettmann = df_suggestion_kettmann.sort_values(by=['points'],
df_suggestion_kettmann
```

Out[65]:

	country	description	designation	points	price	province	region_1	region_2	taster_
121888	Spain	Fondillón represents the pinnacle of red Spani...	Casta Diva Fondillón Sweet	97	88.0	Levante	Alicante	None	M Scha
4521	US	This is a beautiful display of Durell's greatn...	Durell Vineyard Origin	95	48.0	California	Sonoma Valley	Sonoma	V E
122508	US	Spicy acidity lies beneath the full-bodied ric...	Durell Vineyard	95	50.0	California	Sonoma Valley	Sonoma	V E
8882	US	Site speaks loudly in this beautifully made wi...	Durell Vineyard	95	65.0	California	Sonoma Coast	Sonoma	V E
88675	US	Succulent and juicy in velvety waves of black-...	Durell Vineyard	94	65.0	California	Sonoma Coast	Sonoma	V E

```
In [66]: df_czerwinski = df_wine.loc[df_wine['taster_name'] == 'Joe Czerwinski']
df_czerwinski = df_czerwinski.reset_index(drop=True)
```

```
In [67]: ► winery_czerwinski = []
winery_czerwinski = df_czerwinski['winery'].value_counts().keys().to_list()
average_winery_czerwinski = []
for a in winery_czerwinski:
    average_winery_czerwinski.append(df_czerwinski[df_czerwinski['winery']
df_winery_czerwinski = pd.DataFrame({'winery':winery_czerwinski, 'points':a
df_winery_czerwinski = df_winery_czerwinski.sort_values(by=['points'], asce
df_winery_czerwinski = df_winery_czerwinski.set_index('winery')
```

```
In [68]: ► variety_czerwinski = []
variety_czerwinski = df_czerwinski['variety'].value_counts().keys().to_list
average_variety_czerwinski = []
for a in variety_czerwinski:
    average_variety_czerwinski.append(df_czerwinski[df_czerwinski['variety'
df_variety_czerwinski = pd.DataFrame({'variety':variety_czerwinski, 'points
df_variety_czerwinski = df_variety_czerwinski.sort_values(by=['points'], as
df_variety_czerwinski = df_variety_czerwinski.set_index('variety')
```

```
In [69]: df_suggestion_czerwinski = getSuggestion(df_winery_czerwinski, df_variety_c
df_suggestion_czerwinski = df_suggestion_czerwinski.sort_values(by=['points
df_suggestion_czerwinski
```

Out[69]:

	country	description	designation	points	price	province	region_1	region_2	taster
48906	US	This is the smoothest, most polished of Standi...	Mayflower	95	80.0	California	Anderson Valley	None	
22516	US	Crushed blackberry with dried leaves, burned c...	None	94	75.0	California	Templeton Gap District	Central Coast	Ki
70312	US	This young Pinot Noir is difficult and challen...	MoDa	93	90.0	California	Anderson Valley	None	
70307	US	A distinguished Pinot Noir, dry, crisp in acid...	Triangle Block Day Ranch	93	75.0	California	Anderson Valley	None	
11442	France	One of a series of wines made from individual ...	Parcelle Jeanneton No ZB100	91	27.0	Southwest France	Bergerac Sec	None	Rog

```
In [70]: df_sullivan = df_wine.loc[df_wine['taster_name'] == 'Sean P. Sullivan']
df_sullivan = df_sullivan.reset_index(drop=True)
```

```
In [71]: ► winery_sullivan = []
winery_sullivan = df_sullivan['winery'].value_counts().keys().to_list()
average_winery_sullivan = []
for a in winery_sullivan:
    average_winery_sullivan.append(df_sullivan[df_sullivan['winery'] == a].
df_winery_sullivan = pd.DataFrame({'winery':winery_sullivan, 'points':avera
df_winery_sullivan = df_winery_sullivan.sort_values(by=['points'], ascendin
df_winery_sullivan = df_winery_sullivan.set_index('winery')
```

```
In [72]: ► variety_sullivan = []
variety_sullivan = df_sullivan['variety'].value_counts().keys().to_list()
average_variety_sullivan = []
for a in variety_sullivan:
    average_variety_sullivan.append(df_sullivan[df_sullivan['variety'] == a
df_variety_sullivan = pd.DataFrame({'variety':variety_sullivan, 'points':av
df_variety_sullivan = df_variety_sullivan.sort_values(by=['points'], ascend
df_variety_sullivan = df_variety_sullivan.set_index('variety')
```

```
In [73]: df_suggestion_sullivan = getSuggestion(df_winery_sullivan, df_variety_sulli
df_suggestion_sullivan = df_suggestion_sullivan.sort_values(by=['points'],
df_suggestion_sullivan
```

Out[73]:

	country	description	designation	points	price	province	region_1	region_2	taste
1556	US	The flagship wine from Quilceda Creek offers e...	None	99	125.0	Washington	Columbia Valley (WA)	Columbia Valley	Pau
45799	US	The 2006 flagship Cabernet from Quilceda Creek...	None	98	125.0	Washington	Columbia Valley (WA)	Columbia Valley	Pau
125829	US	A bold, muscular, striking wine with nerves of...	None	98	135.0	Washington	Columbia Valley (WA)	Columbia Valley	Pau
55723	US	This feels slightly softer in the mouth than t...	None	96	125.0	Washington	Columbia Valley (WA)	Columbia Valley	Pau
125834	US	Quilceda Creek's Red Mountain Cabernet Sauvign...	Galitzine Vineyard	96	110.0	Washington	Red Mountain	Columbia Valley	Pau

```
In [74]: df_iijima = df_wine.loc[df_wine['taster_name'] == 'Anna Lee C. Iijima']
df_iijima = df_iijima.reset_index(drop=True)
```

```
In [75]: ► winery_iijima = []
winery_iijima = df_iijima['winery'].value_counts().keys().to_list()
average_winery_iijima = []
for a in winery_iijima:
    average_winery_iijima.append(df_iijima[df_iijima['winery'] == a].points)
df_winery_iijima = pd.DataFrame({'winery':winery_iijima, 'points':average_w
df_winery_iijima = df_winery_iijima.sort_values(by=['points'], ascending=False)
df_winery_iijima = df_winery_iijima.set_index('winery')
```

```
In [76]: ► variety_iijima = []
variety_iijima = df_iijima['variety'].value_counts().keys().to_list()
average_variety_iijima = []
for a in variety_iijima:
    average_variety_iijima.append(df_iijima[df_iijima['variety'] == a].points)
df_variety_iijima = pd.DataFrame({'variety':variety_iijima, 'points':average_w
df_variety_iijima = df_variety_iijima.sort_values(by=['points'], ascending=False)
df_variety_iijima = df_variety_iijima.set_index('variety')
```

```
In [77]: df_suggestion_iijima = getSuggestion(df_winery_iijima, df_variety_iijima, '
df_suggestion_iijima = df_suggestion_iijima.sort_values(by=['points'], asce
df_suggestion_iijima
```

Out[77]:

	country	description	designation	points	price	province	region_1	region_2	taster_
41835	France	Even more concentrated than Huët's superb 2009...	Cuvée Constance 500ml	99	159.0	Loire Valley	Vouvray	None	Roger
41838	France	This will be a wonderful wine. Already, it is ...	Le Mont Première Trie Moelleux	97	69.0	Loire Valley	Vouvray	None	Roger
41839	France	A majestic, magisterial wine, powered by super...	Clos du Bourg Première Trie Moelleux	97	69.0	Loire Valley	Vouvray	None	Roger
41852	France	With so much intensity, botrytis and concentra...	None	95	84.0	Loire Valley	Quarts de Chaume	None	Roger
41848	France	A wine that just hints at its enormous future....	Clos du Bourg Moelleux	95	49.0	Loire Valley	Vouvray	None	Roger

```
In [78]: df_gordon = df_wine.loc[df_wine['taster_name'] == 'Jim Gordon']
df_gordon = df_gordon.reset_index(drop=True)
```

```
In [79]: ► winery_gordon = []
winery_gordon = df_gordon['winery'].value_counts().keys().to_list()
average_winery_gordon = []
for a in winery_gordon:
    average_winery_gordon.append(df_gordon[df_gordon['winery'] == a].points)
df_winery_gordon = pd.DataFrame({'winery':winery_gordon, 'points':average_w
df_winery_gordon = df_winery_gordon.sort_values(by=['points'], ascending=False)
df_winery_gordon = df_winery_gordon.set_index('winery')
```

```
In [80]: ► variety_gordon = []
variety_gordon = df_gordon['variety'].value_counts().keys().to_list()
average_variety_gordon = []
for a in variety_gordon:
    average_variety_gordon.append(df_gordon[df_gordon['variety'] == a].points)
df_variety_gordon = pd.DataFrame({'variety':variety_gordon, 'points':average_w
df_variety_gordon = df_variety_gordon.sort_values(by=['points'], ascending=False)
df_variety_gordon = df_variety_gordon.set_index('variety')
```



```
In [81]: df_suggestion_gordon = getSuggestion(df_winery_gordon, df_variety_gordon, '
df_suggestion_gordon = df_suggestion_gordon.sort_values(by=['points'], asce
df_suggestion_gordon
```

Out[81]:

	country	description	designation	points	price	province	region_1	region_2	ta
89728	France	This latest incarnation of the famous brand is...	Cristal Vintage Brut	100	250.0	Champagne	Champagne	None	
36528	France	This is a fabulous wine from the greatest Cham...	Brut	100	259.0	Champagne	Champagne	None	
1557	US	A stupendous Pinot Noir, showing how beautiful...	Precious Mountain Vineyard	99	94.0	California	Sonoma Coast	Sonoma	
47429	US	This expresses the greatness of its vintage an...	Hirsch Vineyard	99	75.0	California	Sonoma Coast	Sonoma	
109428	France	With its gold color and mature, toasty flavors...	Cuvée Sir Winston Churchill Brut	98	305.0	Champagne	Champagne	None	

```
In [82]: df_krebiehl = df_wine.loc[df_wine['taster_twitter_handle'] == '@AnneInVino']
df_krebiehl = df_krebiehl.reset_index(drop=True)
```

```
In [83]: winery_krebiehl = []
winery_krebiehl = df_krebiehl['winery'].value_counts().keys().to_list()
average_winery_krebiehl = []
for a in winery_krebiehl:
    average_winery_krebiehl.append(df_krebiehl[df_krebiehl['winery'] == a].
df_winery_krebiehl = pd.DataFrame({'winery':winery_krebiehl, 'points':avera
df_winery_krebiehl = df_winery_krebiehl.sort_values(by=['points'], ascendin
df_winery_krebiehl = df_winery_krebiehl.set_index('winery')
```

```
In [84]: ► variety_krebiehl = []
          variety_krebiehl = df_krebiehl['variety'].value_counts().keys().to_list()
          average_variety_krebiehl = []
          for a in variety_krebiehl:
              average_variety_krebiehl.append(df_krebiehl[df_krebiehl['variety'] == a
df_variety_krebiehl = pd.DataFrame({'variety':variety_krebiehl, 'points':av
df_variety_krebiehl = df_variety_krebiehl.sort_values(by=['points'], ascend
df_variety_krebiehl = df_variety_krebiehl.set_index('variety')
```

```
In [85]: ► df_suggestion_krebiehl = getSuggestionTwitter(df_winery_krebiehl, df_variet
df_suggestion_krebiehl = df_suggestion_krebiehl.sort_values(by=['points'],
df_suggestion_krebiehl
```

Out[85]:

	country	description	designation	points	price	province	region_1	region_2
39286	Italy	A perfect wine from a classic vintage, the 200...	Masseto	100	460.0	Tuscany	Toscana	N
79104	Italy	Even better than the highly acclaimed 2001 vin...	Masseto	99	250.0	Tuscany	Toscana	N
39287	Italy	Here's a "wow" wine you won't easily forget. M...	Messorio	99	320.0	Tuscany	Toscana	N
118061	US	A tremendous achievement in Merlot. The vineya...	None	98	60.0	California	Mount Veeder	N
109407	Austria	A beautiful wine that is almost like nectar. I...	Zwischen den Seen Nummer 11 Trockenbeerenauslese	97	103.0	Burgenland	None	N

```
In [86]: ► df_buzzeo = df_wine.loc[df_wine['taster_name'] == 'Lauren Buzzeo']
          df_buzzeo = df_buzzeo.reset_index(drop=True)
```

```
In [87]: ► winery_buzzeo = []
winery_buzzeo = df_buzzeo['winery'].value_counts().keys().to_list()
average_winery_buzzeo = []
for a in winery_buzzeo:
    average_winery_buzzeo.append(df_buzzeo[df_buzzeo['winery'] == a].points)
df_winery_buzzeo = pd.DataFrame({'winery':winery_buzzeo, 'points':average_winery_buzzeo})
df_winery_buzzeo = df_winery_buzzeo.sort_values(by=['points'], ascending=False)
df_winery_buzzeo = df_winery_buzzeo.set_index('winery')
```

```
In [88]: ► variety_buzzeo = []
variety_buzzeo = df_buzzeo['variety'].value_counts().keys().to_list()
average_variety_buzzeo = []
for a in variety_buzzeo:
    average_variety_buzzeo.append(df_buzzeo[df_buzzeo['variety'] == a].points)
df_variety_buzzeo = pd.DataFrame({'variety':variety_buzzeo, 'points':average_variety_buzzeo})
df_variety_buzzeo = df_variety_buzzeo.sort_values(by=['points'], ascending=False)
df_variety_buzzeo = df_variety_buzzeo.set_index('variety')
```

```
In [89]: df_suggestion_buzzeo = getSuggestion(df_winery_buzzeo, df_variety_buzzeo, '
df_suggestion_buzzeo = df_suggestion_buzzeo.sort_values(by=['points'], asce
df_suggestion_buzzeo
```

Out[89]:

	country	description	designation	points	price	province	region_1	region_2	tast
39313	US	Delicious aromas of cherry- crusted beef, cocon...	Bilancio	94	36.0	California	Monterey County	Central Coast	†
121252	US	It's no surprise that the Daou brothers blend ...	Mayote	94	85.0	California	Adelaida District	Central Coast	†
85804	US	Made from 75% Syrah with the rest Cabernet Sau...	The Evil Twin	93	65.0	Washington	Red Mountain	Columbia Valley	
68682	US	This bottling blends Syrah and Viognier from C...	Odeon	92	55.0	California	Central Coast	Central Coast	†
106879	US	There's a potpourri of wildly distinct and ama...	None	92	38.0	California	Santa Cruz Mountains	Central Coast	†

```
In [90]: df_kostrzewa = df_wine.loc[df_wine['taster_name'] == 'Susan Kostrzewa']
df_kostrzewa = df_kostrzewa.reset_index(drop=True)
```

```
In [91]: ► winery_kostrzewa = []
winery_kostrzewa = df_kostrzewa['winery'].value_counts().keys().to_list()
average_winery_kostrzewa = []
for a in winery_kostrzewa:
    average_winery_kostrzewa.append(df_kostrzewa[df_kostrzewa['winery'] ==
df_winery_kostrzewa = pd.DataFrame({'winery':winery_kostrzewa, 'points':ave
df_winery_kostrzewa = df_winery_kostrzewa.sort_values(by=['points'], ascend
df_winery_kostrzewa = df_winery_kostrzewa.set_index('winery')
```

```
In [92]: ► variety_kostrzewa = []
variety_kostrzewa = df_kostrzewa['variety'].value_counts().keys().to_list()
average_variety_kostrzewa = []
for a in variety_kostrzewa:
    average_variety_kostrzewa.append(df_kostrzewa[df_kostrzewa['variety'] =
df_variety_kostrzewa = pd.DataFrame({'variety':variety_kostrzewa, 'points':
df_variety_kostrzewa = df_variety_kostrzewa.sort_values(by=['points'], asce
df_variety_kostrzewa = df_variety_kostrzewa.set_index('variety')
```

```
In [93]: df_suggestion_kostrzewa = getSuggestion(df_winery_kostrzewa, df_variety_kos
df_suggestion_kostrzewa = df_suggestion_kostrzewa.sort_values(by=['points'])
df_suggestion_kostrzewa
```

Out[93]:

	country	description	designation	points	price	province	region_1	region_2	tast
90912	Hungary	An intoxicating bouquet of beeswax, honeysuckl...	Aszú 6 Puttonyos	96	125.0	Tokaji	None	None	Jef
25822	Hungary	Aromas of caramelized pineapple, honey and aca...	Aszú 5 Puttonyos	93	50.0	Tokaji	None	None	Jef
84262	South Africa	A blend of Cabernet Sauvignon, Merlot, Petit V...	Signature	93	102.0	Stellenbosch	None	None	
43706	South Africa	The 2007 vintage was generally a great one for...	Signature	93	95.0	Stellenbosch	None	None	
12000	US	Dark gold in color, this very sweet dessert wi...	Home Farm Lipton-Daniel Vineyard	92	30.0	California	Dry Creek Valley	Sonoma	

```
In [94]: df_desimone = df_wine.loc[df_wine['taster_name'] == 'Mike DeSimone']
df_desimone = df_desimone.reset_index(drop=True)
```

```
In [95]: winery_desimone = []
winery_desimone = df_desimone['winery'].value_counts().keys().to_list()
average_winery_desimone = []
for a in winery_desimone:
    average_winery_desimone.append(df_desimone[df_desimone['winery'] == a].
df_winery_desimone = pd.DataFrame({'winery':winery_desimone, 'points':avera
df_winery_desimone = df_winery_desimone.sort_values(by=['points'], ascendin
df_winery_desimone = df_winery_desimone.set_index('winery')
```

```
In [96]: ► variety_desimone = []
variety_desimone = df_desimone['variety'].value_counts().keys().to_list()
average_variety_desimone = []
for a in variety_desimone:
    average_variety_desimone.append(df_desimone[df_desimone['variety'] == a]
df_variety_desimone = pd.DataFrame({'variety':variety_desimone, 'points':av
df_variety_desimone = df_variety_desimone.sort_values(by=['points'], ascend
df_variety_desimone = df_variety_desimone.set_index('variety')
```

```
In [97]: ► df_suggestion_desimone = getSuggestion(df_winery_desimone, df_variety_desim
df_suggestion_desimone = df_suggestion_desimone.sort_values(by=['points'],
df_suggestion_desimone
```

Out[97]:

	country	description	designation	points	price	province	region_1	region_2	taste
100001	France	Chapoutier's selections of the best parcels of...	Le Méal Ermitage	98	150.0	Rhône Valley	Hermitage	None	Roq
100002	France	Jean-Louis, the son of Gérard Chave, is now in...	None	97	125.0	Rhône Valley	Hermitage	None	Roq
108649	Italy	This extraordinary wine put Cabernet Franc on ...	Paleo Rosso	97	130.0	Tuscany	Toscana	None	
121937	Spain	Ethereal aromas of baking spices, cinnamon and...	Les Tosses	96	290.0	Catalonia	Priorat	None	Sc
79108	Italy	This Cabernet Franc is beautiful, with incredi...	Paleo	96	105.0	Tuscany	Toscana	None	

```
In [98]: ► df_jenssen = df_wine.loc[df_wine['taster_name'] == 'Jeff Jenssen']
df_jenssen = df_jenssen.reset_index(drop=True)
```

```
In [99]: ► winery_jenssen = []
winery_jenssen = df_jenssen['winery'].value_counts().keys().to_list()
average_winery_jenssen = []
for a in winery_jenssen:
    average_winery_jenssen.append(df_jenssen[df_jenssen['winery'] == a].poi
df_winery_jenssen = pd.DataFrame({'winery':winery_jenssen, 'points':average
df_winery_jenssen = df_winery_jenssen.sort_values(by=['points'], ascending=
df_winery_jenssen = df_winery_jenssen.set_index('winery')
```

```
In [100]: ► variety_jenssen = []
variety_jenssen = df_jenssen['variety'].value_counts().keys().to_list()
average_variety_jenssen = []
for a in variety_jenssen:
    average_variety_jenssen.append(df_jenssen[df_jenssen['variety'] == a].p
df_variety_jenssen = pd.DataFrame({'variety':variety_jenssen, 'points':aver
df_variety_jenssen = df_variety_jenssen.sort_values(by=['points'], ascendin
df_variety_jenssen = df_variety_jenssen.set_index('variety')
```



```
In [101]: df_suggestion_jenssen = getSuggestion(df_winery_jenssen, df_variety_jenssen)
df_suggestion_jenssen = df_suggestion_jenssen.sort_values(by=['points'], as
df_suggestion_jenssen
```

Out[101]:

	country	description	designation	points	price	province	region_1	region_2	taster
49326	Hungary	Surprisingly subtle, yet maddeningly complex, ...	Essencia	94	764.0	Tokaji	None	None	Ani
6575	Hungary	This exuberant Tokaji starts with an enticing ...	Mézes Mály Aszú 6 Puttonyos	94	175.0	Tokaji	None	None	K
88440	Austria	Gorgeously lifted notes of mint and crushed ci...	None	93	14.0	Burgenland	None	None	Kre
14464	Italy	Made in amphorae made from Georgian clay, this...	Classica	91	85.0	Northeastern Italy	Venezia Giulia	None	
61633	Hungary	For \$20, this bargain basement Tokaji is prett...	Mád Cuvée Late Harvest	91	20.0	Tokaji	None	None	Ani

```
In [102]: df_peartree = df_wine.loc[df_wine['taster_name'] == 'Alexander Peartree']
df_peartree = df_peartree.reset_index(drop=True)
```

```
In [103]: winery_peartree = []
winery_peartree = df_peartree['winery'].value_counts().keys().to_list()
average_winery_peartree = []
for a in winery_peartree:
    average_winery_peartree.append(df_peartree[df_peartree['winery'] == a].
df_winery_peartree = pd.DataFrame({'winery':winery_peartree, 'points':avera
df_winery_peartree = df_winery_peartree.sort_values(by=['points'], ascendin
df_winery_peartree = df_winery_peartree.set_index('winery')
```

```
In [104]: ▶ variety_peartree = []
          variety_peartree = df_peartree['variety'].value_counts().keys().to_list()
          average_variety_peartree = []
          for a in variety_peartree:
              average_variety_peartree.append(df_peartree[df_peartree['variety'] == a]
          df_variety_peartree = pd.DataFrame({'variety':variety_peartree, 'points':av
          df_variety_peartree = df_variety_peartree.sort_values(by=['points'], ascend
          df_variety_peartree = df_variety_peartree.set_index('variety')
```

```
In [105]: ▶ df_suggestion_peartree = getSuggestion(df_winery_peartree, df_variety_peart
          df_suggestion_peartree = df_suggestion_peartree.sort_values(by=['points'],
          df_suggestion_peartree
```

Out[105]:

	country	description	designation	points	price	province	region_1	region_2	taster
120990	France	From a beautifully exposed southwest facing vi...	Château Montus La Tyre	96	160.0	Southwest France	Madiran	None	Rog
120989	France	This is the most recent release of this amazin...	Château Montus XL	96	115.0	Southwest France	Madiran	None	Rog
106847	France	Some vines at Bouscassé are 150 years old, whi...	Château Bouscassé Vieilles Vignes	95	50.0	Southwest France	Madiran	None	Rog
15193	France	Produced from vines planted in 1900 (MCM in Ro...	Vitis MCM	95	35.0	Southwest France	Madiran	None	Rog
83863	France	Some vines at Bouscassé are 150 years old, whi...	Château Bouscassé Vieilles Vignes	95	50.0	Southwest France	Madiran	None	Rog

```
In [106]: ▶ df_dykes = df_wine.loc[df_wine['taster_name'] == 'Carrie Dykes']
          df_dykes = df_dykes.reset_index(drop=True)
```

```
In [107]: ► winery_dykes = []
winery_dykes = df_dykes['winery'].value_counts().keys().to_list()
average_winery_dykes = []
for a in winery_dykes:
    average_winery_dykes.append(df_dykes[df_dykes['winery'] == a].points.mean())
df_winery_dykes = pd.DataFrame({'winery':winery_dykes, 'points':average_winery_dykes})
df_winery_dykes = df_winery_dykes.sort_values(by=['points'], ascending=False)
df_winery_dykes = df_winery_dykes.set_index('winery')
```

```
In [108]: ► variety_dykes = []
variety_dykes = df_dykes['variety'].value_counts().keys().to_list()
average_variety_dykes = []
for a in variety_dykes:
    average_variety_dykes.append(df_dykes[df_dykes['variety'] == a].points.mean())
df_variety_dykes = pd.DataFrame({'variety':variety_dykes, 'points':average_variety_dykes})
df_variety_dykes = df_variety_dykes.sort_values(by=['points'], ascending=False)
df_variety_dykes = df_variety_dykes.set_index('variety')
```

```
In [109]: df_suggestion_dykes = getSuggestion(df_winery_dykes, df_variety_dykes, 'Car
df_suggestion_dykes = df_suggestion_dykes.sort_values(by=['points'], ascend
df_suggestion_dykes
```

Out[109]:

	country	description	designation	points	price	province	region_1	region_2	taster
22712	US	A compelling wine. Made from this Bordeaux ble...	Stagecoach Vineyard	94	145.0	California	Napa Valley	Napa	
93705	US	A compelling wine. Made from this Bordeaux ble...	Stagecoach Vineyard	94	145.0	California	Napa Valley	Napa	
36492	US	Rodney Strong's Symmetry is more tannic than t...	Symmetry Red	94	60.0	California	Alexander Valley	Sonoma	
10756	US	A great Bordeaux- style wine that can stand pro...	Symmetry Red	94	55.0	California	Alexander Valley	Sonoma	
115006	Canada	An artful blend of 43% Cab Franc, 28% Merlot, ...	None	93	49.0	British Columbia	Okanagan Valley	None	Paul (

```
In [110]: df_adams = df_wine.loc[df_wine['taster_name'] == 'Fiona Adams']
df_adams = df_adams.reset_index(drop=True)
```

```
In [111]: ► winery_adams = []
winery_adams = df_adams['winery'].value_counts().keys().to_list()
average_winery_adams = []
for a in winery_adams:
    average_winery_adams.append(df_adams[df_adams['winery'] == a].points.mean())
df_winery_adams = pd.DataFrame({'winery':winery_adams, 'points':average_winery_adams})
df_winery_adams = df_winery_adams.sort_values(by=['points'], ascending=False)
df_winery_adams = df_winery_adams.set_index('winery')
```

```
In [112]: ► variety_adams = []
variety_adams = df_adams['variety'].value_counts().keys().to_list()
average_variety_adams = []
for a in variety_adams:
    average_variety_adams.append(df_adams[df_adams['variety'] == a].points.mean())
df_variety_adams = pd.DataFrame({'variety':variety_adams, 'points':average_variety_adams})
df_variety_adams = df_variety_adams.sort_values(by=['points'], ascending=False)
df_variety_adams = df_variety_adams.set_index('variety')
```

```
In [113]: df_suggestion_adams = getSuggestion(df_winery_adams, df_variety_adams, 'Fio
df_suggestion_adams = df_suggestion_adams.sort_values(by=['points'], ascend
df_suggestion_adams
```

Out[113]:

	country	description	designation	points	price	province	region_1	region_2	taster_
39286	Italy	A perfect wine from a classic vintage, the 200...	Masseto	100	460.0	Tuscany	Toscana	None	
45798	US	Tasted in a flight of great and famous Napa wi...	None	100	200.0	California	Napa Valley	Napa	
121885	US	There are incredible aromatics on this Caberne...	Elevation 1147 Estate	99	150.0	California	Napa Valley	Napa	
79104	Italy	Even better than the highly acclaimed 2001 vin...	Masseto	99	250.0	Tuscany	Toscana	None	
39287	Italy	Here's a "wow" wine you won't easily forget. M...	Messorio	99	320.0	Tuscany	Toscana	None	

```
In [114]: df_pickard = df_wine.loc[df_wine['taster_name'] == 'Christina Pickard']
df_pickard = df_pickard.reset_index(drop=True)
```

```
In [115]: ► winery_pickard = []
winery_pickard = df_pickard['winery'].value_counts().keys().to_list()
average_winery_pickard = []
for a in winery_pickard:
    average_winery_pickard.append(df_pickard[df_pickard['winery'] == a].poi
df_winery_pickard = pd.DataFrame({'winery':winery_pickard, 'points':average
df_winery_pickard = df_winery_pickard.sort_values(by=['points'], ascending=
df_winery_pickard = df_winery_pickard.set_index('winery')
```

```
In [116]: ► variety_pickard = []
variety_pickard = df_pickard['variety'].value_counts().keys().to_list()
average_variety_pickard = []
for a in variety_pickard:
    average_variety_pickard.append(df_pickard[df_pickard['variety'] == a].p
df_variety_pickard = pd.DataFrame({'variety':variety_pickard, 'points':aver
df_variety_pickard = df_variety_pickard.sort_values(by=['points'], ascendin
df_variety_pickard = df_variety_pickard.set_index('variety')
```

In [117]:

```
df_suggestion_pickard = getSuggestion(df_winery_pickard, df_variety_pickard)
df_suggestion_pickard = df_suggestion_pickard.sort_values(by=['points'], ascending=False)
df_suggestion_pickard
```

Out[117]:

	country	description	designation	points	price	province	region_1	region_2	tax
89729	France	This new release from a great vintage for Char...	Le Mesnil Blanc de Blancs Brut	100	617.0	Champagne	Champagne	None	F
36529	France	Pure Chardonnay from the walled vineyard in th...	Clos du Mesnil Brut Blanc de Blancs	99	800.0	Champagne	Champagne	None	F
47429	US	This expresses the greatness of its vintage an...	Hirsch Vineyard	99	75.0	California	Sonoma Coast	Sonoma	
98375	France	In this great year for white Burgundy, Bâtard-...	None	99	560.0	Burgundy	Bâtard-Montrachet	None	F
1557	US	A stupendous Pinot Noir, showing how beautiful...	Precious Mountain Vineyard	99	94.0	California	Sonoma Coast	Sonoma	

In [ ]: