WINE FINE,
THANK YOU

좋은 와인을 찾아드립니다

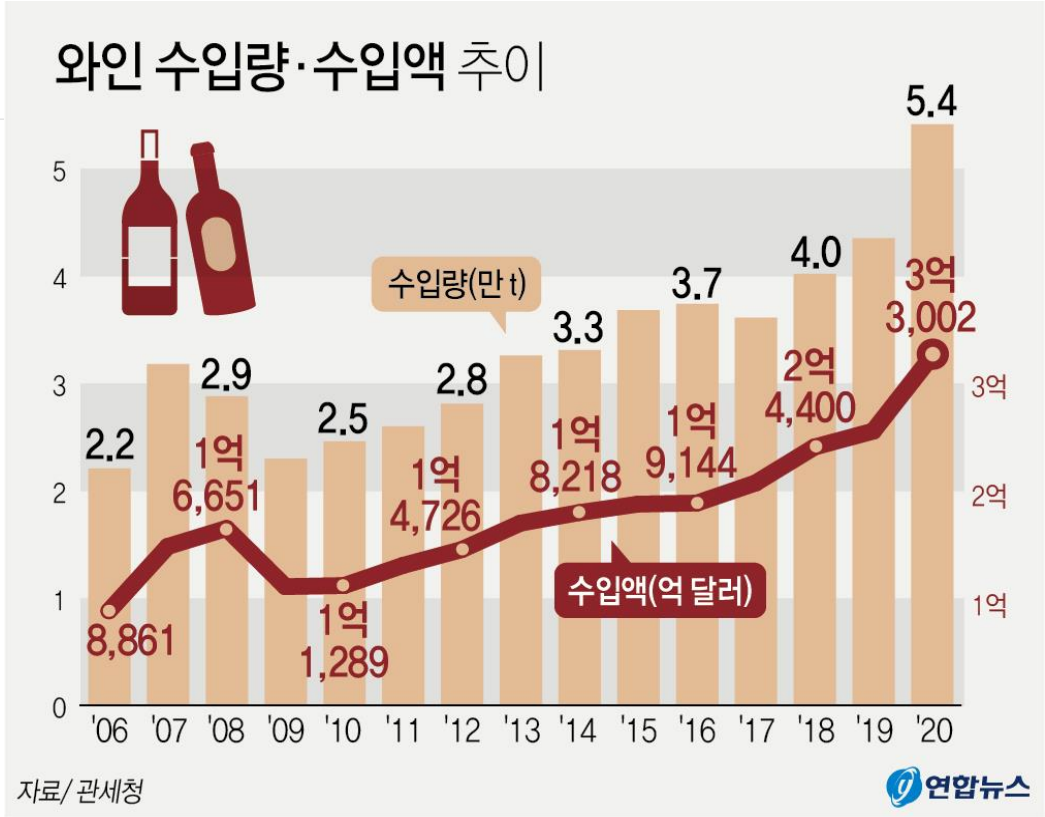
목차

주제 선정 이유

뉴스홈 | 최신기사

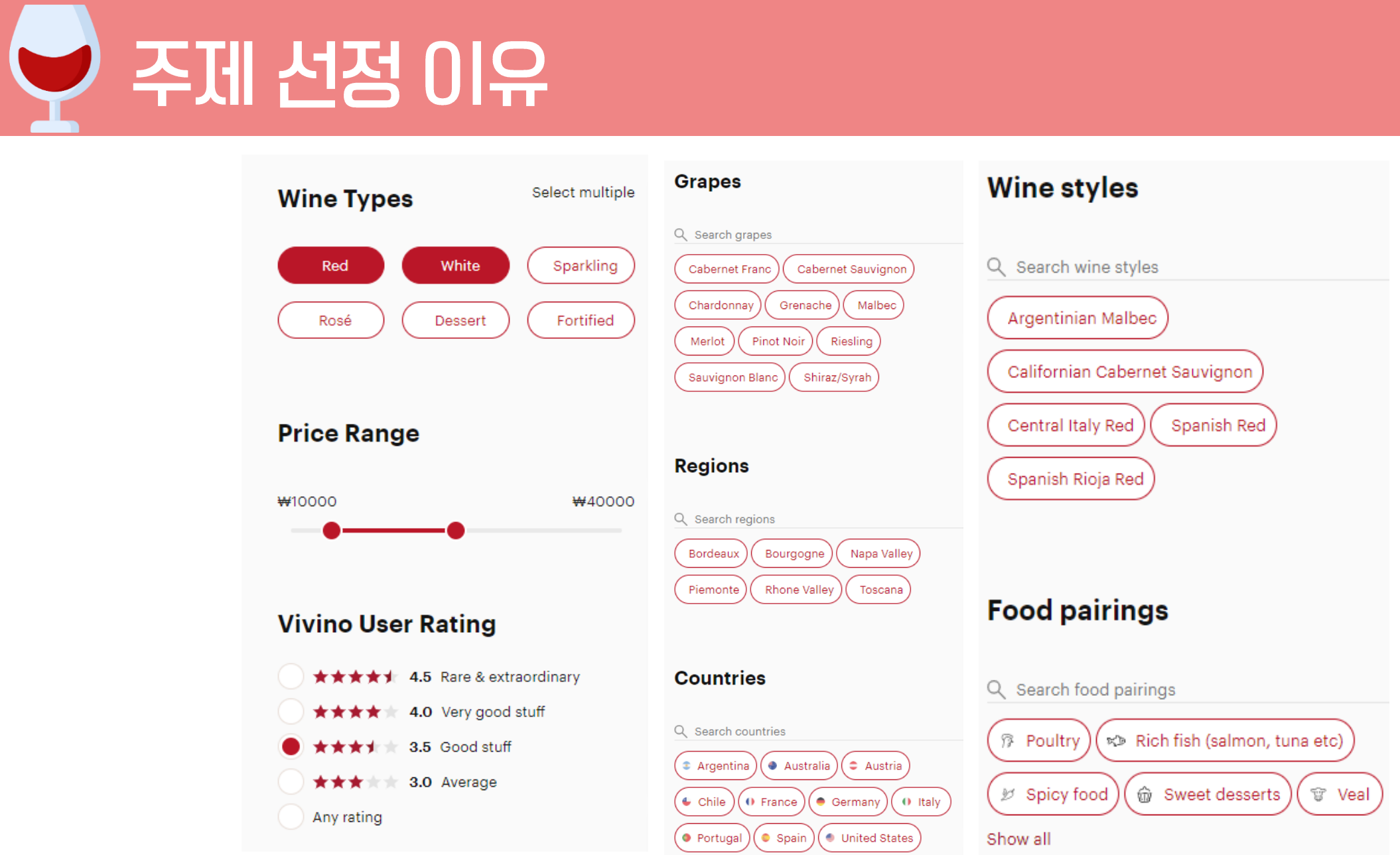
홈술·혼술에 작년 와인 수입 사상 최대…1위 칠레산

송고시간 | 2021-03-04 06:30



김계리 인턴 / 20210304
트위터 @yonhap_graphics 페이스북 tuney.kr/LeYN1

주제 선정 이유


Wine Types
Select multiple
Red
White
Sparkling
Rosé
Dessert
Fortified
Price Range
₩10000
₩40000
Vivino User Rating
4.5 Rare & extraordinary
4.0 Very good stuff
3.5 Good stuff
3.0 Average
Any rating
Grapes
Search grapes
Cabernet Franc
Cabernet Sauvignon
Chardonnay
Grenache
Malbec
Merlot
Pinot Noir
Riesling
Sauvignon Blanc
Shiraz/Syrah
Regions
Search regions
Bordeaux
Bourgogne
Napa Valley
Piemonte
Rhone Valley
Toscana
Countries
Search countries
Argentina
Australia
Austria
Chile
France
Germany
Italy
Portugal
Spain
United States
Wine styles
Search wine styles
Argentinian Malbec
Californian Cabernet Sauvignon
Central Italy Red
Spanish Red
Spanish Rioja Red
Food pairings
Search food pairings
Poultry
Rich fish (salmon, tuna etc)
Spicy food
Sweet desserts
Veal
Show all

주제 선정 이유

Zilliken 2019 Rausch Auslese Riesling (Mosel) GERMANY
This zippy, laser-edged auslese offsets luminous tangerine, white peach and honeydew flavors ...
Editors' Choice SEE FULL REVIEW ▸
98 Points
$104

Dr. Loosen 2014 Erdener Prälat Réserve Alte Reben Dry GG Riesling (Mosel) GERMANY
This remarkable wine displays breathtaking power and elegance. Sourced from the sun-drenched ...
Editors' Choice SEE FULL REVIEW ▸
97 Points
$162

Maximin Grünhäuser 2019 Abtsberg GG Riesling (Mosel)
GERMANY
Struck flint and river rocks introduce this brilliantly steely dry Riesling. Compared ...
Cellar Selection SEE FULL REVIEW ▸
97 Points
$70

Schäfer-Fröhlich 2018 Bockenauer Stromberg GG Dry Gold Cap Riesling (Nahe) GERMANY
Sourced from volcanic soils, this is a smoldering, intensely mineral wine highlighted ...
Cellar Selection SEE FULL REVIEW ▸
97 Points
$125

설계

와인의 퀄리티에 영향을 끼치는 요인의 분석, 인사이트 도출

리뷰 데이터에서 자연어 분석을 통해 와인 특징에 대한 단어들을 얻음

취향 군집화 시도

검색엔진 서비스 개발

분석 및 결과 1. 지도학습

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.4	0.700	0.00	1.9	0.076	11.0	34.0	0.99780	3.51	0.56	9.4	5
1	7.8	0.880	0.00	2.6	0.098	25.0	67.0	0.99680	3.20	0.68	9.8	5
2	7.8	0.760	0.04	2.3	0.092	15.0	54.0	0.99700	3.26	0.65	9.8	5
3	11.2	0.280	0.56	1.9	0.075	17.0	60.0	0.99800	3.16	0.58	9.8	6
4	7.4	0.700	0.00	1.9	0.076	11.0	34.0	0.99780	3.51	0.56	9.4	5

AdaBoost

```
my_max_depth = 9                                        # 고정해 둔다.
my_learn_rate = 0.01                                    # 고정해 둔다.
n_estimators_grid = np.arange(50, 81, 2)
parameters = {'n_estimators': n_estimators_grid}
AB = AdaBoostClassifier(base_estimator=DecisionTreeClassifier(max_depth=my_max_depth), learning_rate=my_learn_rate)
#instantiate an estimator.
gridCV = GridSearchCV(AB, param_grid=parameters, cv=10, n_jobs = -1)
gridCV.fit(X_train, Y_train)
best_n_estim = gridCV.best_params_['n_estimators']
```

```
print("AdaBoost best n estimator : " + str(best_n_estim))
```

```
AdaBoost best n estimator : 76
```

```
AB_best = gridCV.best_estimator_                        # 교차검증의 결과인 최적의 학습객체 사용.
Y_pred = AB_best.predict(X_test)
print( "AdaBoost best accuracy : " + str(np.round(metrics.accuracy_score(Y_test,Y_pred),3)))
```

```
AdaBoost best accuracy : 0.64
```

xg boost

```
#xg boost
my_max_depth = 4                                        # 고정해 둔다.
my_learn_rate = 0.1                                     # 고정해 둔다.
n_estimators_grid = np.arange(300, 601, 100)
parameters = {'n_estimators': n_estimators_grid}
XGBC = XGBClassifier(max_depth=my_max_depth, learning_rate=my_learn_rate)
#instantiate an estimator.
gridCV = GridSearchCV(XGBC, param_grid=parameters, cv=10, n_jobs = -1)
gridCV.fit(X_train, Y_train)
best_n_estim = gridCV.best_params_['n_estimators']
```

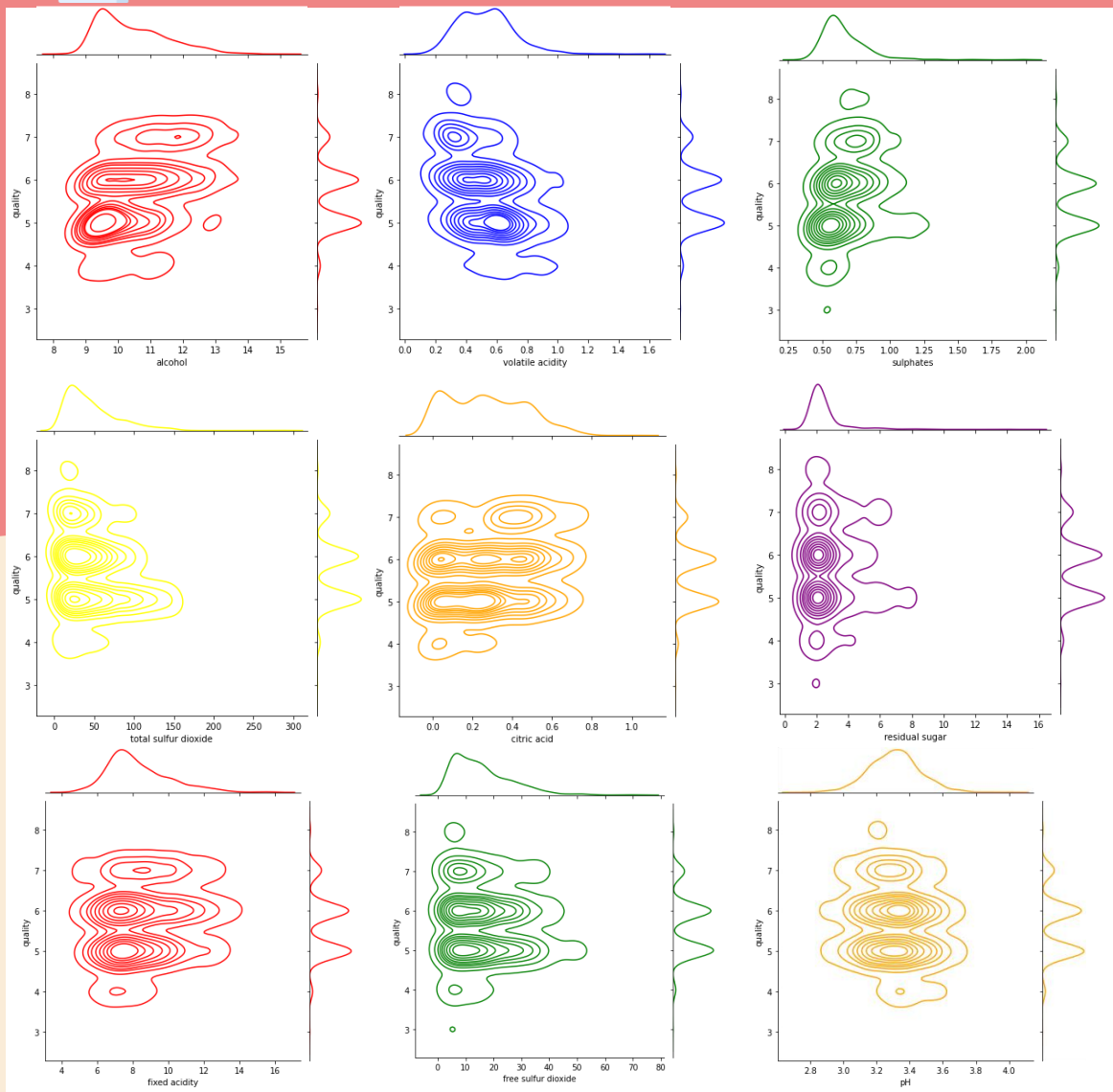
```
print("XGBoost best n estimator : " + str(best_n_estim))
```

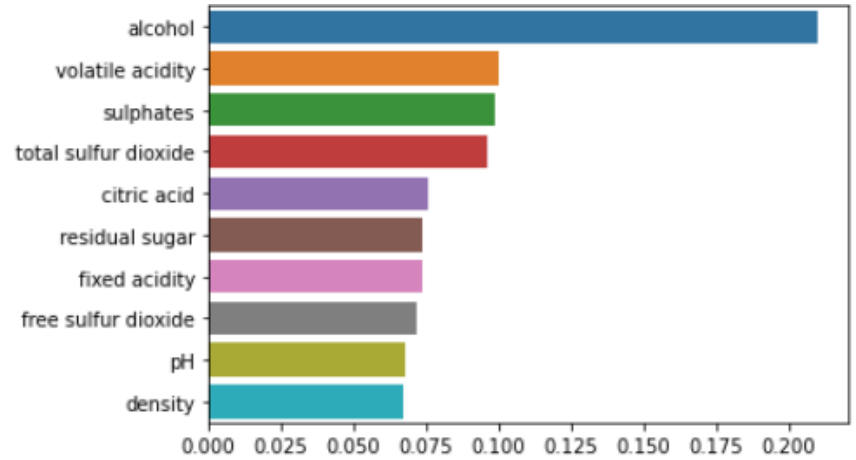
```
XGBoost best n estimator : 400
```

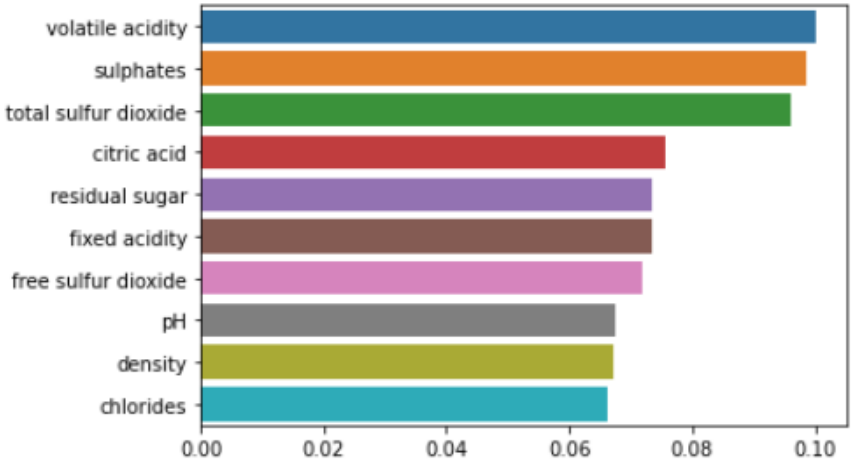
```
XGBC = XGBClassifier(n_estimators = best_n_estim, learning_rate = 0.1, max_depth = 4, random_state=123)
XGBC.fit(X_train, Y_train)
Y_pred = XGBC.predict(X_test)
print( "XGBoost accuracy : " + str(np.round(metrics.accuracy_score(Y_test,Y_pred),3)))
```

```
XGBoost accuracy : 0.638
```

분석 및 결과 1. 지도학습


quality
alcohol
volatile acidity
sulphates
total sulfur dioxide
citric acid
residual sugar
fixed acidity
free sulfur dioxide
pH


alcohol
volatile acidity
sulphates
total sulfur dioxide
citric acid
residual sugar
fixed acidity
free sulfur dioxide
pH
density
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
0.200


volatile acidity
sulphates
total sulfur dioxide
citric acid
residual sugar
fixed acidity
free sulfur dioxide
pH
density
chlorides
0.00
0.02
0.04
0.06
0.08
0.10

분석 및 결과 2. 자연어 분석

```
cleaned=[]
for sentence in df.description:
    pre=re.sub(r'₩₩', ' ', sentence)
    pre=re.sub('fruity', 'fruit', pre)
    pre=re.sub('vanilla', 'oak', pre)
    pre=nltk.word_tokenize(pre)
    pre = [x.lower() for x in pre]
    pre = [x for x in pre if x not in stopwords.words('english')]
    pre = [lemmatizer.lemmatize(x) for x in pre]
    pre = [stemmer.stem(x) for x in pre]
    pre=[re.sub(fruit, 'fruit', x) for x in pre]
    cleaned.append(pre)
```

분석 및 결과 2. 자연어 분석

```
remove_list = ['also','charact','come','still','drink','years','feel',
               'flavor','like','made','show','hint','offer','give','wine',
               'finish','2017','2018','2019','acid','although','well','estat','develop','year','need','readi','age',
              'note']
wine_feature_variety = ['black','cabernet','merlot','pinot','red','white','sauvignon']
```

```
my_vectorizer = TfidfVectorizer(max_features = 50, min_df=0.1,max_df=0.8,
                                stop_words = stopwords.words('english')+remove_list+wine_feature_variety)
X = my_vectorizer.fit_transform(df3['cleaned']).toarray()
print(my_vectorizer.get_feature_names())
```

```
['aroma', 'balanc', 'blend', 'bodi', 'chocol', 'dark', 'dri', 'firm', 'full', 'herb', 'nose', 'note', 'oak', 'palat', 'pepper', 'ric
h', 'ripe', 'soft', 'spice', 'structur', 'tannin']
```

분석 및 결과 3. 군집화

```
my_km = KMeans(n_clusters = 3, random_state = 123)
my_km.fit(X)
my_centroids = my_km.cluster_centers_                # 개개 군집의 중심점.
my_cluster_labels = my_km.labels_
```

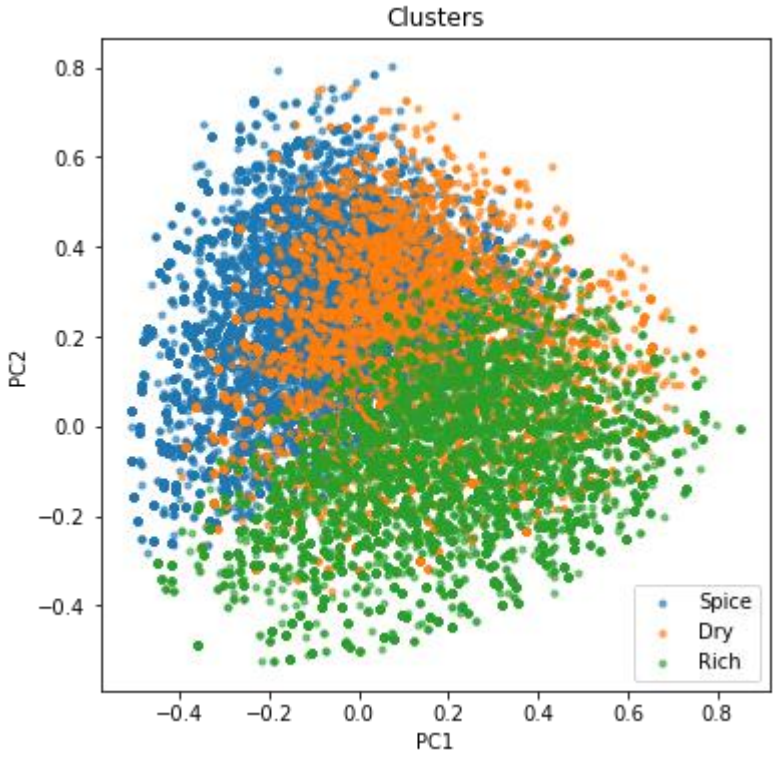
```
for i in range(len(my_centroids)):
    print(i)
    print([my_vectorizer.get_feature_names()[x] for x in np.argsort(my_centroids[i])[-5:]])
```

```
0
['spice', 'note', 'palat', 'aroma', 'fruit']
1
['acid', 'aroma', 'tannin', 'fruit', 'dri']
2
['rich', 'ripe', 'acid', 'tannin', 'fruit']
```

```
# PCA 차원축소 (2차원).
my_pca = PCA(n_components = 2)
transformed_comps = my_pca.fit_transform(X)          # Transformed 된 좌표.
df_transformed_comps = pd.DataFrame(data = transformed_comps, columns = ['PC1', 'PC2'])
df_transformed_comps=df_transformed_comps.join(pd.Series(my_cluster_labels, name='cluster_label'))
```

```
my_names = {0: 'Spice', 1: 'Dry' ,2: 'Rich'}

plt.figure(figsize = (6,6))
for a_cluster_n, df_small in df_transformed_comps.groupby('cluster_label'):
    plt.scatter('PC1', 'PC2', data = df_small, label = my_names[a_cluster_n], s = 10, alpha=0.6 )
plt.xlabel('PC1')
plt.ylabel('PC2')
plt.title('Clusters')
plt.legend(loc=4)
plt.show()
```



분석 및 결과 4. BOW

```
lstFind = {"feature": ["ripe", "green", "fruit"], "price":[30, 50]}
def find(argDicTarget, argDfData):
    dfTmp = df1[  argDfData["feature"].apply(lambda x : isInStr(argDicTarget["feature"], x))
                & argDfData["price"].apply(lambda x : int(x) > lstFind["price"][0])
                & argDfData["price"].apply(lambda x : int(x) < lstFind["price"][1])
               ].copy()

    dfTmp["numFt"] = dfTmp["feature"].apply(lambda x: cntFeature(argDicTarget["feature"], x))
    dfTmp["numCb"] = dfTmp["variety"].apply(lambda x: cntFeature(["Cabernet"], x))

    return(dfTmp.sort_values(by=["numFt", "price", "points", "numCb"], ascending=[False, True, False, False])[['country', 'descripti
                  'price', 'province', 'region_1', 'taster_name','title', 'variety', "numCb", 'winery']].rename(columns={"region_1":"

find(lstFind, df1)
```

	country	description	numFt	designation	points	price	province	region	taster_name	title	variety	numCb	winery
123013	Austria	Grassy notes of green conference pear with inc...	3	Gamlitzer	90	31.0	Südsteiermark	NaN	Anne Krebiehl MW	Sattlerhof 2014 Gamlitzer Sauvignon Blanc (Süd...	Sauvignon Blanc	0	Sattlerhof
53565	Italy	Fruity aromas of ripe orchard fruit and citrus...	3	Nadin Dry	89	31.0	Veneto	Valdobbiadene Prosecco Superiore	Kerin O'Keefe	Foss Marai 2015 Nadin Dry (Valdobbiadene Pros...	Glera	0	Foss Marai
34986	US	Blue fruit, raspberry, violet, dark chocolate,...	3	NaN	92	32.0	Washington	Walla Walla Valley (WA)	Sean P. Sullivan	Trust 2014 Syrah (Walla Walla Valley (WA))	Syrah	0	Trust
46432	US	Blended with 4% Viognier and 1% Grenache, this...	3	8 Clones Red Willow Vineyard	92	32.0	Washington	Yakima Valley	Sean P. Sullivan	Eight Bells 2012 8 Clones Red Willow Vineyard ...	Syrah	0	Eight Bells
89836	US	Blended with 4% Viognier and 1% Grenache, this...	3	8 Clones Red Willow Vineyard	92	32.0	Washington	Yakima Valley	Sean P. Sullivan	Eight Bells 2012 8 Clones Red Willow Vineyard ...	Syrah	0	Eight Bells

결론

- 와인의 맛에 영향을 주는 요소에 대한 인사이트 도출
- 소믈리에들의 맛표현을 참고해 와인 특징 키워드 제공
- 머신러닝으로는 자연어 감성 분석을 통한 예측이 힘듦

발전방향

- 딥러닝 활용: 와인 리뷰 분석을 통한 점수 예측
- 딥러닝 활용: n-gram을 통한 와인 특성에 대한 감성 예측

lemmatize 에서 유용한 단어 선별(Tfidf)

```
lstStopWord = []
vectMy = TfidfVectorizer(max_features = 50, token_pattern=r"(?u)₩b₩w₩w+-₩w+₩b", stop_words = lstStopWord )

X = vectMy.fit_transform(df1['NGram']).toarray()

print(vectMy.get_feature_names())
```

```
['aroma-flavor', 'bake-spice', 'berri-flavor', 'berri-fruit', 'black-cherri', 'black-currant', 'black-frui
t', 'black-pepper', 'black-plum', 'bodi-wine', 'bright-acid', 'cabernet-franc', 'cabernet-sauvignon', 'che
rri-flavor', 'cherri-fruit', 'crisp-acid', 'drink-2017', 'drink-2018', 'drink-2020', 'easi-drink', 'finish
-drink', 'firm-tannin', 'flavor-finish', 'french-oak', 'fresh-acid', 'fruit-aroma', 'fruit-flavor', 'fruit
-wine', 'full-bodi', 'green-appl', 'medium-bodi', 'nose-palat', 'palat-deliv', 'palat-offer', 'petit-verdo
t', 'pinot-noir', 'readi-drink', 'red-berri', 'red-cherri', 'red-currant', 'red-fruit', 'ripe-fruit', 'sau
vignon-blanc', 'stone-fruit', 'tropic-fruit', 'white-peach', 'white-pepper', 'wine-offer', 'wine-show', 'w
ood-age']
```

들어주셔서 감사합니다