데이터 전처리

INDEX

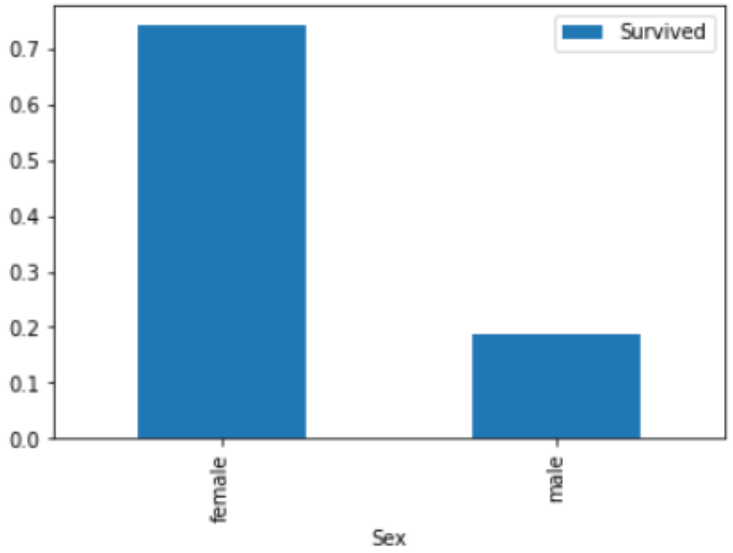
1. 데이터 가치 파악

1. 데이터 가치파악

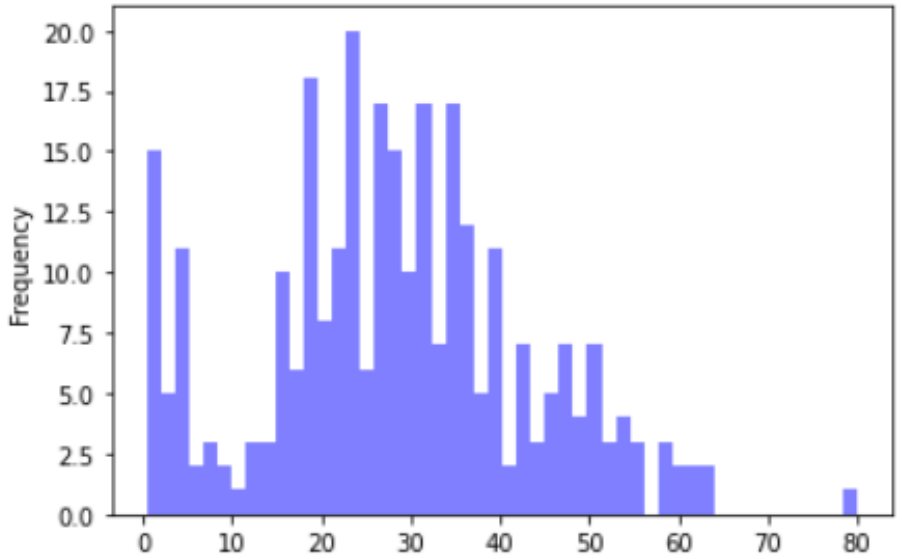
	승객 아이디	생존여부	사회,경제적 지위	승객이름	성별	나이	동승가족	동승부모	티켓번호	가격	객실구역	선착장
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583	NaN	Q
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625	E46	S
7	8	0	3	Palsson, Master. Gosta Leonard	male	2.0	3	1	349909	21.0750	NaN	S
8	9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0	0	2	347742	11.1333	NaN	S
9	10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	1	0	237736	30.0708	NaN	C

1. 데이터 가치파악

```
sex_pivot = train.pivot_table(index="Sex", values="Survived")
sex_pivot.plot.bar()
plt.show()
#성별
```

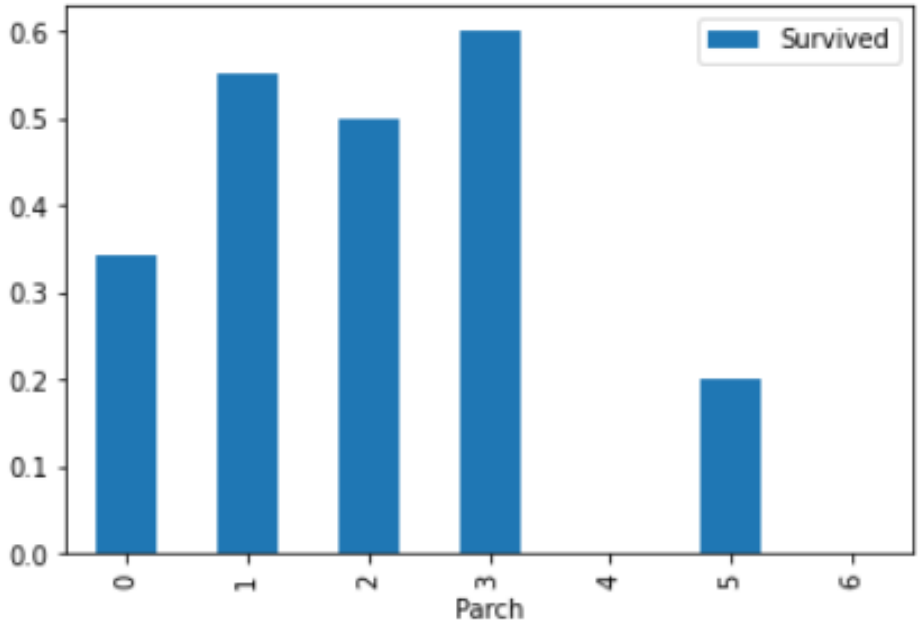



```
survived = train[train["Survived"]==1]
survived["Age"].plot.hist(alpha=0.5, color='blue',bins=50)
plt.show()
```



1. 데이터 가치파악

```
1 Parch_pivot = train.pivot_table(index="Parch", values="Survived")
2 Parch_pivot.plot.bar()
3 plt.show()
```



2. 결측치 처리

2. 결측치 처리

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  891 non-null    int64
 1   Survived     891 non-null    int64
 2   Pclass       891 non-null    int64
 3   Name         891 non-null    object
 4   Sex          891 non-null    object
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64
 7   Parch        891 non-null    int64
 8   Ticket       891 non-null    object
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object
 11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

2. 결측치 처리

	A	B	C	D	E	F	G	H	I	J	K	L
1	Passenger	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
2	1	0	3	Braund, M	male	22	1	0	A/5 21171	7.25		S
3	2	1	1	Cumings,	female	38	1	0	PC 17599	71.2833	C85	C
4	3	1	3	Heikkinen	female	26	0	0	STON/O2.	7.925		S
5	4	1	1	Futrelle, M	female	35	1	0	113803	53.1	C123	S
6	5	0	3	Allen, Mr.	male	35	0	0	373450	8.05		S
7	6	0	3	Moran, M	male		0	0	330877	8.4583		Q

평균값 +표준 편차

중간값

최빈값

2. 결측치 처리

```
#Age
mean = train_temp['Age'].mean()  # 평균
std = train_temp['Age'].std()   # 표준편차
is_null_cnt = train_temp['Age'].isnull().sum() #결측치 갯수 측정
rand_age = np.random.randint(mean-std, mean+std, size=is_null_cnt)  #평균 + 표준편차

#Null인 경우에는 나이의 평균 분포 내에 있는 임의의 값들 선택

rand_temp = train_temp['Age'].copy()
rand_temp[np.isnan(rand_temp)] = rand_age
train_temp['Age'] = rand_temp
```

3. 데이터 변환

3. 데이터 변환

Out[144]:

	Survived	Pclass	Sex	Age	SibSp	Fare	Embarked
0	0	3	male	22.0	1	7.2500	S
1	1	1	female	38.0	1	71.2833	C
2	1	3	female	26.0	0	7.9250	S
3	1	1	female	35.0	1	53.1000	S
4	0	3	male	35.0	0	8.0500	S

```
# Embarked
```

```
1 #Fare
2 train_temp['Fare'] = train_temp['Fare'].fillna(0).astype(int)
3 #int 형식으로 바꿔서 뒷자리 잘라줬음
```

```
train_temp["Embarked"] = train_temp["Embarked"].map(ports)
```

3. 데이터 변환

	Survived	Pclass	Sex	Age	SibSp	Fare	Embarked
0	0	3	0	22.0	1	7	0
1	1	1	1	38.0	1	71	1
2	1	3	1	26.0	0	7	0
3	1	1	1	35.0	1	53	0
4	0	3	0	35.0	0	8	0

3. 데이터 변환

```
X_train = train_temp.drop("Survived", axis=1)
y_train = train_temp['Survived']
X_train.head()

#LightGBM

lgbm_clf = LGBMClassifier(n_estimators=100)
lgbm_clf.fit(X_train,y_train)
print("score: ",round(lgbm_clf.score(X_train, y_train)*100,2))

#RandomForest
from sklearn.ensemble import RandomForestClassifier

clf = RandomForestClassifier(n_estimators=100)
clf.fit(X_train,y_train)
print("score: ",round(clf.score(X_train, y_train)*100,2))

#XGBoost
from xgboost import XGBClassifier
import xgboost as xgb

xgb_clf = XGBClassifier(n_estimators=100)
xgb_clf.fit(X_train,y_train)
print("score: ",round(xgb_clf.score(X_train, y_train)*100,2))
```

3. 데이터 변환

```
score:  92.93
```
LightGBM

```
score:  96.52
```
Random Forest

```
C:₩Users₩JaeKyeom₩anaconda3₩lib₩site-packages₩xgboost₩sklearn.py:888: UserWarning: The use of label encoder in XGBClassifier is deprecated and will be removed in a future release. To remove this warning, do the following: 1) Pass option use_label_encoder=False when constructing XGBClassifier object; and 2) Encode your labels (y) as integers starting with 0, i.e. 0, 1, 2, ..., [num_class - 1].
  warnings.warn(label_encoder_deprecation_msg, UserWarning)

[01:02:47] WARNING: C:/Users/Administrator/workspace/xgboost-win64_release_1.3.0/src/learner.cc:1061: Starting in XGBoost 1.3.0, the default evaluation metric used with the objective 'binary:logistic' was changed from 'error' to 'logloss'. Explicitly set eval_metric if you'd like to restore the old behavior.
score:  94.95
```
XGBoost

4. Q & A

Q & A

감사합니다 ㅎㅎ