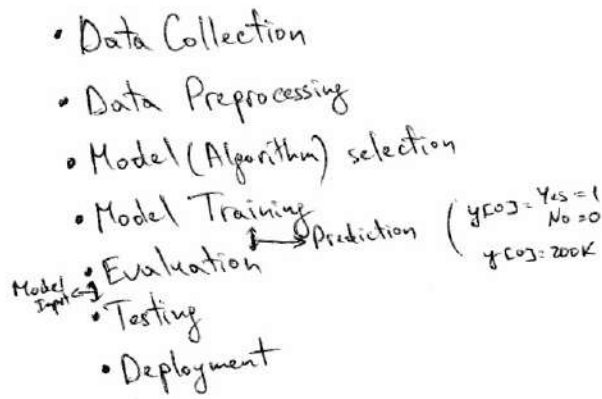


ML model structure



- Linear Regression

- Linear Family
- Baseline Model
- Regression (Continues)
 - Players goals
 - Not injury
- Linear Equation
 - X - input
 - y - Output
 - Beta - parameter
 - epsilon - ERROR

→ Linear Regression (parameter)

- Simple LR
- Multiple LR

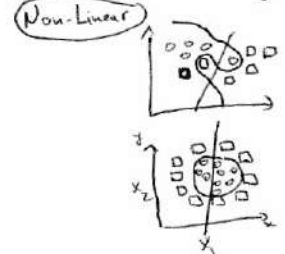
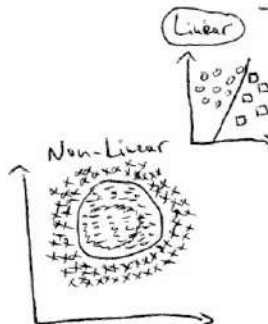
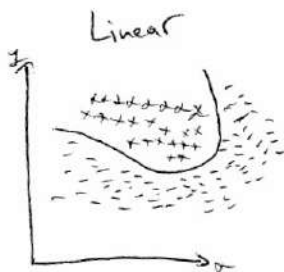
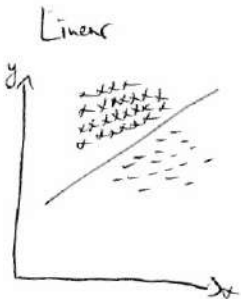
Epsilon != 0 (epsilon hechqachon "0" bo'lmaydi)

~ Kuchli tomonlari

- Tez va sodda
- Kichikroq dasturlar b/n ishlatishda oson
- Baseline model

~ Kuchli tomonlari?

- Kam chipim
- Complex (non-linear) larni o'rganish oson
- Yot elementlarda tez o'zgaruvchan (stabilizatsiya)
- Katta dasturlarda yaxshi ishlatiladi.



1) # Linear Regression analizi

from sklearn.linear_model import LinearRegression

lr = LinearRegression()

lr

2) # Training

X = df.drop('target column', axis=1)

y = df['target column']

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

lr.fit(X_train, y_train)

3) # Prediction

y_pred/y_test = lr.predict(X_test)

y_pred[0]

4) # Evaluation

from sklearn.metrics

Evaluation - quyidagilarni baxolaydi:

prediction gahvatlik yaxshiligi / model ishonchligi / Generalization? Overfitting?

Evaluation → Classification: → Accuracy score, Classification Report

 Evaluation → Regression: → Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), R² Score (Coefficient of Determination)

4) # Evaluation

```
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
```

```
mae = mean_absolute_error(y_test, y_pred)
```

```
mse = mean_squared_error(y_test, y_pred)
```

```
r2 = r2_score(y_test, y_pred)
```