**혁신성장 청년인재 집중양성 사업 - 1기**

|  |
| --- |
| **서울·부산 지역**  **아파트 실 거래가 예측** |

혁신성장 청년인재

딥러닝 기반 핵심산업별 빅데이터 분석 전문가 과정(C반)

**듀란듀란**

**유선우 민유진 함윤선 장정호**

**2019년 1월 31일**

**목 차**

**1. 프로젝트 개요**  **1**

1.1 프로젝트 기획 배경 및 목표 1

1.2 구성원 및 역할 2

1.3 프로젝트 추진 일정 2

**2. 프로젝트 현황**  **3**

2.1 개요 3

2.2 데이터 수집 4

2.3 데이터 전처리 5

2.4 Feature Engineering 8

**3. 프로젝트 분석 결과**  **9**

3.1 주요 모델링 9

3.2 OLS 9

3.3 Random Forest 11

3.4 Deep Learning 13

3.5 GBM(Gradient Boosting Algorithm) 16

**4.** 프로젝트 결과 및 개선사항 **23**

4.1 프로젝트 결과 및 개선사항 23

4.2 기대 효과 24

**5. 분석 후기**  **26**

**1. 프로젝트 개요**

1.1 프로젝트 기획 배경 및 목표

프로젝트의 시작은 단순한 호기심에서 시작되었습니다. 각 지역 출신의 팀원들은 ‘누구 집이 더 비싼가?’에 대한 이야기를 나누고 있었기 때문입니다. 그러다 각 지역 아파트의 시세를 보고 ‘아파트 가격은 어떻게 정해지는가?’ 에 대해 궁금해졌습니다.

아파트 가격에는 수많은 변수들이 영향을 미칩니다. 물가나 소득, 금리와 같이 경제 사이클과 밀접한 거시적 경제 및 금융변수들부터 ‘‘학군’와 ‘지역의 평균 소득’, 사람들의 사회적 지위와 관련된 부촌 지역과 같은 심리/사회적 요소 등이 높습니다. 게다가 지하철이나 공원, 조망권의 유무와 같은 지역 위치 정보 변수들도 지대한 영향을 끼치며 마지막으로 아파트 자체의 정보들도 있습니다.

저희는 아파트 거래가격에 영향을 주는 수 많은 변수들 중, 현재 정보 접근이 가능하고, 분석의 객관성을 담보할 수 있는 변수들을 선정했습니다. 바로 지역 정보와 아파트 자체 정보 데이터입니다. 이 자료들은 금융정보처럼 계량적이면서도 데이터의 접근성이 용이하고, 심리적 요소들은 제외하여 분석의 객관성을 담보할 수 있다고 판단했습니다. 따라서 위의 데이터를 가지고 빅데이터 분석을 진행하여 아파트들의 기존 실 거래가를 확인하고 나아가 향후 미래 실 거래가 예측까지 할 수 있는 프로젝트를 진행하게 되었습니다.

이번 프로젝트의 목표는 크게 두 가지로 나눌 수 있습니다.

먼저, 아파트 단지 정보, 매매 정보 등 다양한 비 금융 데이터를 기반으로 보다 정밀하게 아파트 실 거래가를 예측하는 모델을 만드는 것입니다. 기존의 금융데이터와 시계열 데이터를 제외하고도 분석 모델을 만들 수 있다면 그 자체로 성공적이라고 생각했기 때문입니다.

다음으로 모델의 예측 오차를 줄여 단순히 분석 모델을 구현하는 것에서 그치는 것이 아니라 기존 가치 산정방법들을 대체하거나 보완할 수 있도록 성능을 극대화 시키는 것입니다.

우리는 분석 모델의 성능을 측정하는 지표로 RMSE(Root Mean Squared Error)를 선택하여 모델의 예측 값과 실제 값의 차이를 확인하고자 했습니다. 저희 듀란듀란 팀은 거래가격의 차이를 최소화 하여 모델의 성능을 극대화하는 것을 목표로 삼았습니다.

한 달여의 프로젝트를 통해 2006~2018년까지의 아파트 실 거래가와 아파트 정보를 분석, 아파트 가격을 예측하는 모델을 구성하고, 이후 검증 데이터를 통해 분석 모델의 성능을 확인하고 개선하는 과정을 진행했습니다. 프로젝트 종료 후 저희의 목표였던 아파트 가격을 예측하는 모델을 구현하는 데 성공했습니다.

본 프로젝트는 금융/재무정보들을 분석하여 아파트 가치를 산정하는 기존의 방법에서 벗어나 보다 새로운 관점에서 가격을 측정하는 신선한 시도가 될 것입니다.

1.2 구성원 및 역할

|  |  |  |  |
| --- | --- | --- | --- |
| 이름 | 전공 | 역할 | 구현 부분 |
| 유선우 | 경영학 | 팀장 | 1.일정표 작성 및 진행 상황 보고(총괄)  2.기본 OLS기법 적용  3.Machine Learning(Random Forest) 기법 적용  4.추가 변수 검색 및 적용(park)  5.Boosting 기법 분석 및 모델링 적용(LightGBM) |
| 장정호 | 경제학 | 팀원 | 1.딥러닝 방법론 분석 및 모델링 적용(keras)  2.Boosting 기법 분석 및 모델링 적용(LightGBM)  3.Light GBM 전담 모델 개선  4.병렬 모델 구성(Ensemble 모형구축을 통한 예측률 개선)  6.최종 결과물 발표 |
| 함윤선 | 경영정보학 | 팀원 | 1.추가 변수 검색 및 적용 시도(school, subway)  2.Boosting 기법 분석 및 모델링 적용(XGBoost)  3.K-Fold validation 보완  4.Presentation 자료 작성 |
| 민유진 | 산업경영공학 | 팀원 | 1.EDA 분석 및 시각화  2.딥러닝 방법론 분석 및 모델링 적용(keras)  3.Boosting 기법 분석 및 모델링 적용(XGBoost)  4.XG Boost 전담 모델 개선  5.코드 개선 및 예측률 개선 |

1.3 프로젝트 추진 일정

|  |  |  |  |
| --- | --- | --- | --- |
| 구분 | 기간 | 활동 | 비고 |
| 주제 선정 | 12/28(금) ~ | 프로젝트 팀 구성 | 4인 팀  (유선우, 장정호, 함윤선, 민유진) |
| 사전 기획 | 12/28(금) ~ 1/4(금) | 프로젝트 주제 선정 | 아파트 실 거래가 예측 |
| 1/5(토) | 멘토 결정 | 허주용 멘토님 |
| 1/8(화) ~ 1/18(금) | 프로젝트 사전 회의 | 주제 구체화 및 토론 |
| 1/19 (토) | 팀 별 착수 보고 |  |
| PJT 수행 | 1/19(토) ~ 2/27(수) | 프로젝트 수행 |  |
| 1/21(월) ~ 2/1(금) | EDA 및 Feature Engineering | 시각화 및 추가 변수 정리 |
| 2/2(토) ~ 2/6(수) | 설날 연휴 |  |
| 2/7(목) ~ 2/23(토) | Model Selection  Regularization  모델 개선 | OLS, Random Forest, Keras, Light GBM, XGBoost |
| 2/25(월) | 최종 결과물 제출 | 보고서 및 최종 발표 PPT 제출 |
| 2/27(수) | 최종 보고 리허설 |  |
| 2/28(목) | 팀별 최종 발표  (구축 완료 보고) |  |

**2. 프로젝트 현황**

2.1 개요

본 프로젝트에서 프로젝트 과정은 다음의 순서를 거칩니다. [표1]

[표1. Data Science Process]

|  |  |
| --- | --- |
| 순서 | 단계 |
| 1 | Project Scoping(문제 정의) |
| 2 | Data Collection(데이터 수집) |
| 3 | EDA(탐색적 데이터 분석) |
| 4 | Data Preprocessing(데이터 전처리) |
| 5 | Feature Engineering(변수 가공 및 생성) |
| 6 | Modeling |
| 7 | Evaluation |
| 8 | Project Delivery / Insights(프로젝트 결과 및 개선) |

본 프로젝트에서 문제 정의는 ‘아파트 가격에 어떠한 변수들이 영향을 미치는가?’ 였습니다. 이 문제를 가시화하기 위해서 저희는 비 재무/금융데이터로 아파트 가격을 분석하고 예측하는 모델을 구성하기로 하였습니다. 다음으로 데이터 수집 단계에서는 비 재무/금융데이터의 수집을 위해 서울 지역 아파트 거래 데이터와 위치 정보데이터를 수집하였습니다.

2.2 데이터 수집

프로젝트 진행을 위해 필요했던 데이터는 바로 아파트 단지 정보와 매매 정보가 필요했습니다. 단지 정보는 아파트 자체 변수를 의미하는데 아파트 준공 년도, 공급 면적, 공용 사용면적에서부터 방의 개수, 냉난방 종류, 주차장 정보가 이에 해당합니다. 우리는 흔히 최근에 지어진 아파트인지, 복도식인지, 아파트 평수는 얼마나 되는 지가 실제 아파트 구매에 고려하는 만큼 해당 데이터를 구할 필요가 있었습니다.

다음으로 매매정보에는 아파트 거래일자(년월일)와 거래 가격이 해당됩니다. 아파트 거래 일자는 시계열 데이터로서 시간의 흐름에 따라 아파트 거래가격이 변동하는 만큼 중요한 변수라고 판단했습니다. 또한 정부의 정책과 그 당시의 경제상황이 거래가격에 영향을 미치는 만큼 거래 일자 데이터 속에 이러한 시계열 흐름이 반영될 것이기 때문입니다. 이는 금융/재무 데이터를 반영하지 않고도 분석 모델 성능을 담보할 수 있는 중요한 요인이었습니다. 거래 가격은 분석에서 결과변수로서 분석 모델을 학습하고 검증하는 데에 필수적입니다.

위치 정보 데이터도 저희의 중요한 변수였습니다. 아파트 가격에는 아파트 자체적 요소도 중요하지만 주변의 시설위치 또한 거래여부 및 가격에 지대한 영향을 끼칩니다. 거주하는 아파트 주변에 교통이 편리한지(지하철/버스 노선 여부), 자녀가 있는 경우 교육인프라가 구성되어 있는지(학교), 산책할 공간의 여부(공원), 조망권의 여부 등이 바로 거래가격에 반영되고, 구매를 결정하는 요소로 작용합니다.

저희가 수집한 데이터는 다음과 같습니다.[표2]

[표2. 데이터 수집 정보 및 현황]

|  |  |  |  |
| --- | --- | --- | --- |
| 순서 | 수집 데이터 | 비고 | 출처 |
| 1 | 아파트 단지 정보 | 위치정보 ,준공 년도, 공급면적, 주차장 정보 등 | 국토교통부 실 거래가 공개시스템([rt.molit.go.kr](http://rt.molit.go.kr/)) |
| 2 | 매매 정보 | 거래 일자, 거래가격 등 | 국토교통부 실 거래가 공개시스템([rt.molit.go.kr](http://rt.molit.go.kr/)) |
| 3 | 지하철 노선 및 환승역 정보 | 일반 역, 환승역 여부, 위치 정보 | 서울열린데이터 광장(data.seoul.go.k/) |
| 4 | 서울 시내 초, 중, 고등학교 현황 | 학교의 종류, 위치 정보 | 서울열린데이터 광장(data.seoul.go.k/) |
| 5 | 서울시 공원 현황 | 공원의 종류, 위치 정보 | 공공데이터포털(data.go.kr) |

2.3 데이터 전처리

데이터의 세부 설명은 다음과 같습니다. [표3]

[표3. 데이터 변수 정보 및 설명]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **변수** | **내용** | **Type** | **특징** | **변수 아이디어** |
| key | Primary key |  |  |  |
| apartment\_id | 아파트 ID | int64 | 숫자로 되어 있는 아파트 식별번호이지만, 엄밀히 식별번호다 | 100만 개의 데이터셋 모두를 확인하기 어려운 경우에 아파트 단위로 묶어서 활용해 볼 것. |
| city | Seoul: 1 | int64 | 1. 서울 : 1004009개 |  |
| transaction\_year\_month | 실 거래가  년,월 : yyyymm | int64 | 1. train\_year : 2006~2018  2. test\_year : 2009~2018  3. month : 01~12 | Year, Month Feature값으로 분할 |
| transaction\_date | 실 거래가 발생일  ( dd ~ dd) | object | 1~10', '11~20', '21~30', '21~31', '21~28' | 숫자화할 경우에는 앞자리 수만으로 구분을 지을 것.  '1~10'은 '초반', '11~20'은 '중반',  '21~30', '21~31', '21~28'의 경우엔 '후반'으로  동일하게 처리. |
| year\_of\_completion | 아파트  준공년도, 입주년 | int64 | 1962,1966,1968,1969,1970,1971,1972,1974, 1975,1976,1977,  1978,1979,1980,1981,1982,1983,1984,1985,1986,1987,1988,  1989,1990,1991,1992,1993,1994,1995,1996,1997,1998,1999,  2000,2001,2002,2003,2004,2005,2006,2007,2008,2009,2010,  2011,2012,2013,2014,2015,2016,2017,2018 | 준공 년도로부터 현재까지의 년도로 아파트 연식을 확인하기.  49년 이후 아파트의 경우에는 거래가가 확연히 떨어지므로 구분해보기.  아파트 연 식에 따른 카테고리화를 진행해 보기. |
| exclusive\_use\_area | 전용면적 | float64 | supply\_area-public\_area = exclusive\_area | supply\_area와 연관지어 public\_area 확인 가능 |
| floor | 층 | int64 | max : 80층  75% : ~14층 | 층수에는 '로얄층'에 해당하는 층이 있음.  로얄층을 따로 구분지어 볼 것.  혹은 층 수별로 로얄층에 변동이 있으므로 비율화로 나타내보기. |
| latitude | latitude | float64 | 위도 | 거리 관련 변수 생성에 활용. |
| longitude | longitude | float64 | 경도 | 거리 관련 변수 생성에 활용. |
| address\_ by\_law | 법정동 코드 | int64 | (앞2자리: 시/도, 앞5자리: 시/군/구, 앞8자리: 읍/면/동) - 법정동 코드 (10자리)  : 법적주소 - 시도(2)+시군구(3)+읍면동(3)+리(2)  https://www.code.go.kr/stdcode/regCodeL.do | 시/군/구 별 구분 지을 때 기준변수로 활용.  시/군/구 별 실 거래가 평균 구하는 변수로 활용. |
| total\_parking\_capacity\_in\_site | 아파트 단지 내 총 주차대수 | float64 | 결측치 포함되어 있음. 조정 필요.  train + test : 92184 | 1. 간단하게 평균으로 결측치를 채우는 경우(mean)  2. 결측치 부분을 모두 빼버리는 경우  3. 앞전의 자료로 채우는 경우 (ffill)  4. 회귀식을 활용해서 채우는 경우(linear regression) |
| total\_household\_count\_in\_sites | 아파트 단지 내 총세대수 | int64 | min : 99  max : 6864 | 1. 주차와 관련해 세대 당 주차 공간 구하기 |
| apartment\_building\_count\_in\_sites | 아파트 단지 내 총 동수 | int64 | min : 1  max : 124 |  |
| tallest\_building\_in\_sites | 아파트 단지 내 최고층 | float64 | min : 5  max : 80  결측치 포함되어 있음. 조정 필요.  train + test : 10 | 1. floor 변수와 같이 활용해서 loyal floor 변수 생성에 활용하기. |
| lowest\_building\_in\_sites | 아파트 단지 내 최저층 동의 층수 | float64 | min : 1  max : 70  결측치 포함되어 있음. 조정 필요.  train + test : 10 |  |
| heat\_type | 난방방식 : 개별난방(individual), 중앙난방(central), 지역난방(district) | object | 결측치 포함되어 있음. 조정 필요.  train + test : 2045 | 1. 난방 방식에 따른 변화 추이 확인할 것, |
| heat\_fuel | 난방연료 : 도시가스(gas), 열병합(cogeneration) | object | 결측치 포함되어 있음. 조정 필요.  train + test : 9715 | 1. 난방 연료에 따른 변화 추이 확인할 것. |
| room\_id | 평형 ID | int64 | (평형 =(전용면적㎡+ 주거공용면적㎡)\*0.3025))  (1평형=3.3058㎡) | 1. 평당 실 거래가 구하는 변수로 활용하기. |
| supply\_area | 공급면적 | float64 | (= 전용면적 + 주거공용면적) | 1. exclusive\_use\_area 변수와 같이 활용해서 주거공용면적 변수 생성하기. |
| total\_household\_count\_of\_area\_type | 평형의 총세대수 | int64 | 평당 총 세대수 |  |
| room\_count | 방 수 | float64 | max : 8  50% : 3개 | 75% : ~3개  결측치 포함되어 있음. 조정 필요.  train + test : 696 | 1. 방의 개수에 따른 실 거래가 추이 파악하기 |
| bathroom\_count | 욕실 수 | float64 | max : 5개  75% : ~2개  결측치 포함되어 있음. 조정 필요.  train + test : 696 | 1. 욕실 개수에 따른 실 거래가 추이 파악하기 |
| front\_door\_structure | 현관구조 : 복도식(corridor), 계단식(stairway), 복합식(mixed) | object | 결측치 포함되어 있음. 조정 필요.  train + test : 13937 | 1. 구조에 따른 변화 추이 확인할 것. |
| transaction\_real\_price | 실 거래가 (단위: 원) | int64 | 결과 변수 |  |

2.4 Feature Engineering

Feature Engineering은 머신러닝 알고리즘을 작동하기 위해 데이터에 대한 도메인 지식을 활용하여 특징(Feature)을 만들어내는 과정입니다.[[1]](#footnote-1) 다르게 정의하면, 머신러닝 모델을 위한 데이터 테이블의 컬럼(특징)을 생성,선택,가공하는 일련의 과정을 의미합니다. 이것을 하는 이유는 바로 모델의 성능을 높이기 위해서입니다. Feature Engineering은 모델 성능에 미치는 영향이 크기 때문에 머신러닝 응용에 있어서 굉장히 중요한 단계이며, 전문성과 시간, 그리고 비용이 많이 드는 작업입니다.

Feature Engineering 구성에서 저희가 집중했던 부분은 도메인(분야) 전문성 측면입니다. 대표적인 것이 특징 생성(Feature Generation)과 특징 구축(Feature Construction)입니다. 저희는 기존 변수로부터 ‘apart\_49’(준공된 지 49년 이상의 노후 아파트)열과 ‘loyal\_floor’의 여부, ‘address\_by\_law’ 변수로부터 구별, 동별 데이터를 분리하는 등 다양한 특징들을 생성했습니다. 결과적으로는 Feature Engineering을 진행한 모델의 성능이 하지 않은 모델의 성능보다 높음을 알 수 있었습니다.

**3. 프로젝트 분석 결과**

3.1 주요 모델링

본 프로젝트에서는 주어진 데이터를 다양한 분석 알고리즘으로 분석해보고 그 성능을 측정해보고자 했습니다. 그러나 한정된 시간으로 각 팀원이 분석 알고리즘을 나누어 맡아 분석을 진행했습니다. 각 팀원 별 진행한 분석 알고리즘은 다음과 같습니다. [표4]

[표4. 구성원 별 담당 분석 알고리즘]

|  |  |
| --- | --- |
| 팀원 명 | 알고리즘 |
| 유선우 | OLS, RandomForest, lightGBM(GBM) |
| 장정호 | Keras, lightGBM(GBM) |
| 민유진 | XGBoost(GBM) |
| 함윤선 | XGBoost(GBM) |

3.2 OLS

3.2.1 개요

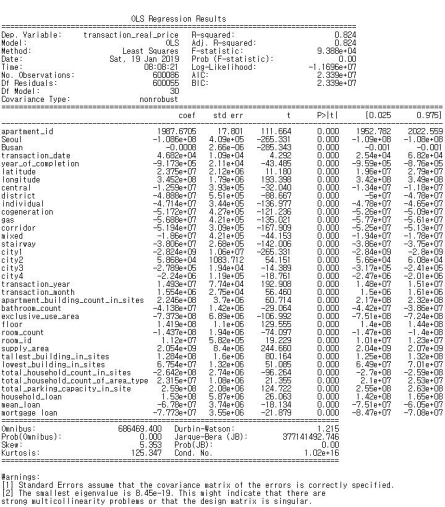
OLS(Ordinary Least Squares)는 선형 회귀를 활용한 추정 방법 중에서 가장 단순하고 많이 쓰이는 방법입니다. 사전 이해를 위해 우선 선형 회귀에 대해서 설명을 하자면, 선형 회귀는 종속 변수 y와 한 개 이상의 독립 변수 x와의 선형 상관 관계를 모델링하는 회귀분석 기법을 의미합니다. 독립 변수의 개수에 따라 한 개인 경우에는 단순 선형 회귀, 둘 이상인 경우에는 다중 선형 회귀로 나눕니다. 선형 예측 함수를 사용해 선형 회귀는 회귀 식을 모델링하며, 알려지지 않은 파라미터를 데이터로부터 추정할 수 있다. 이렇게 만들어진 회귀 식을 선형 모델이라고 부릅니다.



선형 회귀에서 파라미터 추정을 위한 다양한 기법들이 개발되었는데, 알고리즘의 연산 복잡도나 데이터 분포에 대한 가정, 변수들의 관계에 대한 가정 등에 따라서 다양성을 띄게 됩니다. 파생된 여러가지 기법들 중에서 개념적으로 가장 단순하고 계산이 간단한 것이 바로 OLS 추정입니다. OLS 기법은 오차의 제곱의 합을 최소화하는 기법입니다.

3.2.2 의의

OLS 기법에 따른 결과물은 가장 간단하고 기본적이기 때문에 모든 분석 기법들과의 비교를 위한 기준점이 되어 줄 수 있다는 데 의의가 있습니다. 또한, 아래의 그림과 같이 R-squared 수치를 통해서 전반적인 정확도를 파악할 수 있으며, 사용된 변수들이 결과값과 어느 정도의 연관성이 있는지 coef 수치를 통해서 확인할 수 있습니다.



OLS 기법의 경우에는 기법 자체로서의 결과물인 RMSE 점수는 낮다고 할 수 없지만, 그 자체로 변수와 모델의 개선점의 기준이 될 수 있다는 면에서 의의가 있습니다.

3.3 Random Forest(Machine Learning)

3.3.1 개요

랜덤 포레스트(Random Forest)는 머신러닝의 한 기법으로, 여러 개의 의사결정나무(Decision Tree)들을 임의적으로 학습하는 방식의 앙상블(Ensemble) 방법을 의미합니다. 랜덤 포레스트 기법은 의사결정나무의 단점을 보완하기 위해 개발된 기법으로, 의사결정나무는 계층적 접근방식이기 때문에 만약 중간에 오류가 발생한다면 다음 단계로 그 오류가 계속 전파되는 특성을 가지고 있습니다. 이러한 단점을 극복하기 위해 배깅(bagging), 임의 노드 최적화와 같은 임의화 기술을 활용해 랜덤 포레스트는 좋은 일반화 성능을 갖게 되었습니다. 상관 관계가 없는 트리들로 포레스트를 구성해 랜덤 포레스트를 학습하게 됩니다.

랜덤 포레스트 방법은 두 가지 단계로 구성됩니다. 첫 번째로 다수의 의사결정나무를 구성하는 학습 단계와 두 번째로 입력 벡터가 들어왔을 때 분류 하거나 예측하는 테스트 단계가 그것입니다. 첫 번째 다수의 의사결정나무를 구성하는 과정에서 배깅과 임의 노드 최적화 두 가지 방법을 활용하게 됩니다.

배깅(Bagging)을 통해 랜덤 포레스트를 학습시키는 과정은 크게 세 단계로 구성됩니다. 먼저 부트스트랩(Bootstrap)방법을 통해 N개의 훈련 데이터 집합을 생성하고, 다음으로 N개의 기초 트리들을 훈련시키며, 마지막으로 트리들을 하나의 분류기(랜덤 포레스트)로 결합합니다. 임의 노드 최적화 학습 방식은 각각의 트리 내에 존재하는 노드마다 분할 함수를 이용하여 각 노드에서의 정보 획득량이 최대가 되도록 최적화하는 방법입니다.

랜덤 포레스트의 가장 핵심적인 특징은 임의성(Randomness)에 의해 서로 조금씩 다른 특성을 갖는 트리들로 구성이 된다는 것입니다. 임의성은 각 트리들의 예측들이 비상관화되게 하며, 결과적으로 일반화 성능을 향상시키게 됩니다. 또한 포레스트가 노이즈가 포함된 데이터에 대해서도 강인하게 만들어주는 효과가 있습니다.



3.4 Deep Learning

3.4.1 개요

인공신경망(Artificial neural network) 모형은 층, 연결강도, 전이함수, 학습 알고리즘으로 이루어진 데이터 처리 시스템으로서 입력 자료값과 해당 출력자료 값들을 통해 가중치들이 반복적으로 조정되어 결국 입력 및 출력자료간의 관계가 학습되는 구조입니다.

신경망의 기본개념은 생물학적 뇌 구조와 동작을 모방하여 계산 모델에 적용한 것입니다.[[2]](#footnote-2) 뇌의 전기화학적 신경신호 전말물질과 전달경로를 통하여 학습시킴으로써 지능적으로 행동할 수 있도록 합니다. 표준적인 신경 네트워크는 뉴런으로 구성되어 있습니다. 인풋 뉴런이 활성화 되어 다른 뉴런에 연결이 되고, 이러한 뉴런의 연결이 반복되면서 학습됩니다.[[3]](#footnote-3)

자료에 대한 표현 능력을 크게 증가시키기 위해 은닉 뉴런수를 증가시키거나, 은닉층을 추가하여 사용하였지만 역전파(backpropagation) 알고리즘이 에러 정보를 출력층에서 입력층 방향으로 전달되면서 점점 사라진다는 문제가 발생하였고, 이에 2006년 사전학습이 포함된 딥 러닝(Deep Learning) 모형을 제시하였습니다.[[4]](#footnote-4) 딥러닝의 경우 기존 인공신경망과 달리 마지막 은닉층까지 충분히 전파가 가능하여 딥러닝 분석을 이용한 금융자료예측에 적용하고 있습니다.[[5]](#footnote-5)

한편 부동산 가격평가측면에서는 전통적으로 선형회귀모형을 활용해왔습니다. 하지만 이 접근은 설명변수와 결과변수 간 선형의 관계를 가정한다는 한계점이 있었습니다. 이러한 한계를 극복하기 위해 다층신경망 모형을 활용하여 문제 해결에 접근해보았습니다.

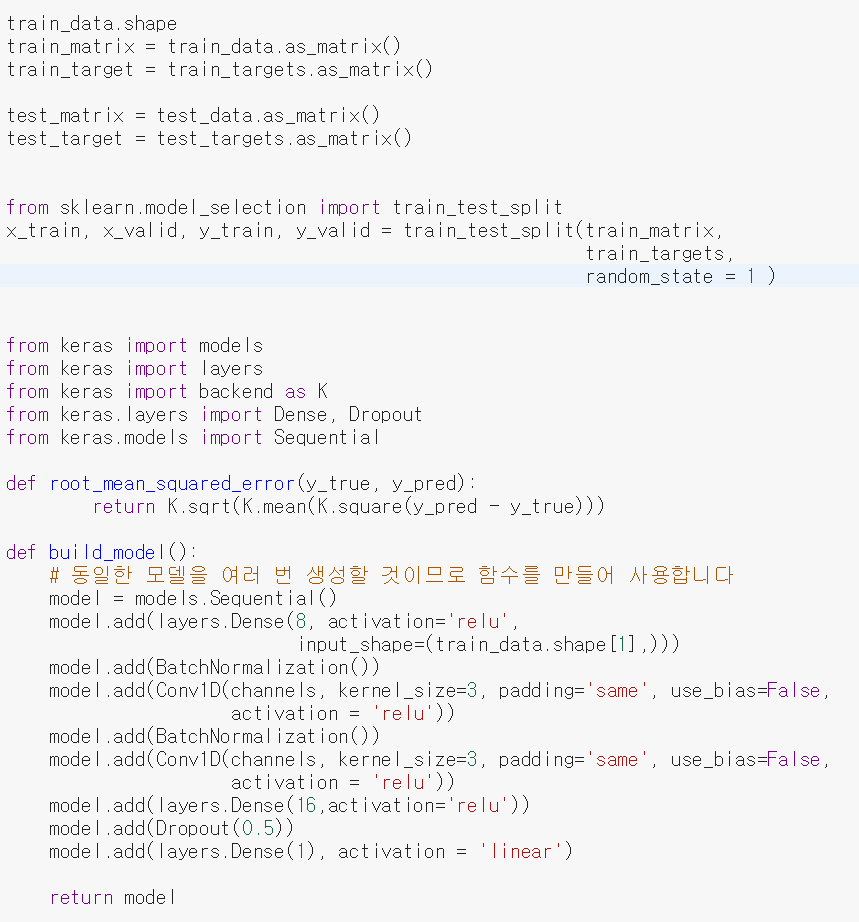
3.4.2 Keras

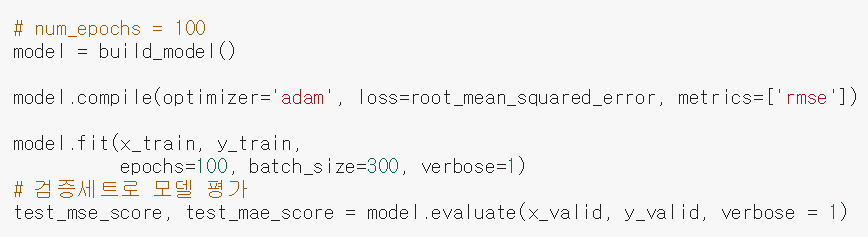
Keras는 Python으로 구현된 쉽고 간결한 딥러닝 라이브러리입니다. 딥러닝 비전문가라도 각자 분야에서 손쉽게 딥러닝 모델을 개발하고 활용할 수 있도록 케라스는 직관적인 API를 제공하고 있습니다. 내부적으로는 텐서플로우 (TensorFlow), 티아노(Theano), CNTK 등의 딥러닝 전용 엔진이 구동되지만 케라스 사용자는 복잡한 내부 엔진을 알 필요는 없습니다. 직관적인 API로 쉽게 다층퍼셉트론 모델, 컨볼루션 신경망 모델, 순환 신경망 모델 또는 이를 조합한 모델은 물론 다중 입력 또는 다중 출력 등 다양한 구성을 할 수 있습니다.

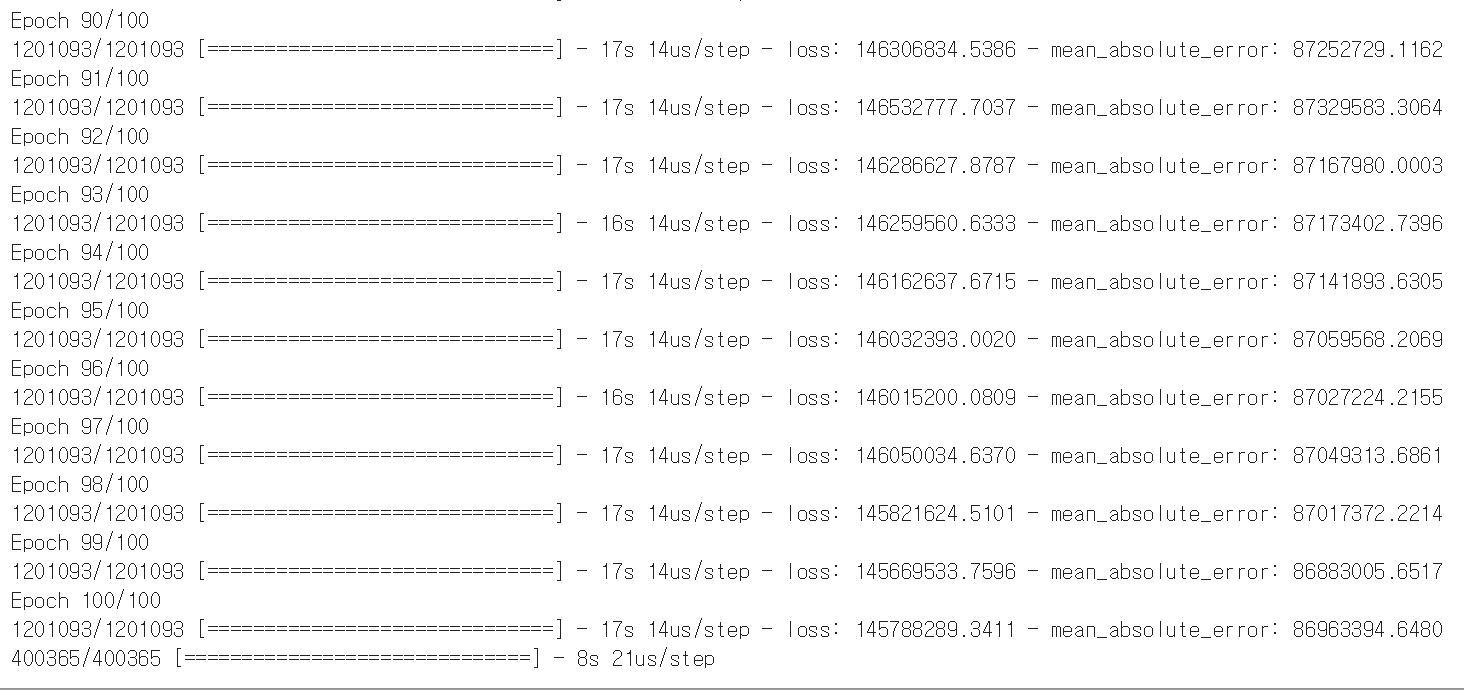
케라스의 주요 특징은 다음과 같습니다. [표5]

[표5. Keras 주요 특징점]

|  |  |  |
| --- | --- | --- |
| **Keras** | 특징 | 비고 |
| 모듈화(Modularity) | 독립적이고 설정 가능하며, 최소한의 제약사항으로 서로 연결될 수 있습니다.  특히, 신경망 층, 비용함수, 최적화/초기화/정규화 기법은 모두 독립적이며, 새로운 모델을 구성하기 위해 모듈을 조합할 수 있습니다. |
| 최소주의 (Minimalism) | 각 모듈은 짧고 간결합니다.  모든 코드는 한 번 훑어보는 것으로도 이해가능합니다.  반복속도와 혁신성에는 다소 떨어질 수 있습니다. |
| 확장성(Extensibilty) | 새로운 클래스나 함수로 모듈을 쉽게 추가할 수 있습니다. |
| Python 기반(Based) | Caffe처럼 별도의 모델 설정 파일이 필요없으며 Python 코드로 모델이 정의됩니다. |





****

3.5 GBM(Gradient Boosting Algorithm)

3.5.1 개요

Gradient Boosting Algorithm(GBM)은 회귀분석 또는 분류분석을 수행할 수 있는 예측모형이며 예측모형의 앙상블 방법론 중 Boosting 계열에 속하는 알고리즘입니다. 이 알고리즘은 Tabular format 데이터에 대한 예측에서 좋은 성능을 보여주고, 머신러닝 알고리즘에서도 예측 성능이 높다고 알려져 있는데, XGBoost, LightGBM, CatBoost들이 GBM으로 구현된 패키지입니다.

GBM의 원리는 다음과 같습니다. 모델 A를 통해 결과값을 예측하고 남은 잔차(residual)을 다시 B라는 모델을 통해 예측합니다. A+B모델을 통해 y를 예측한다면 A보다 나은 B모델을 만들 수 있습니다. 반복을 통해 잔차를 줄이고, Training Set을 잘 설명하는 예측모형이 만들어집니다. 하지만 문제는 이를 통해 Bias를 줄일 수 있어도, 과적합이 일어날 가능성이 있다는 것입니다. 따라서 실제로 GBM을 사용할 때는 Sampling, Penalizing 등의 정규화 기법을 이용하여 개선된 모델을 이용하는 것이 보편적인 방법입니다.

3.5.2 XGBoost

XGBoost란 학습 성능은 좋지만, 수행시간/연산시간 측면에서는 부족한 Gradient Boost를 엄청나게 개선한 것입니다.

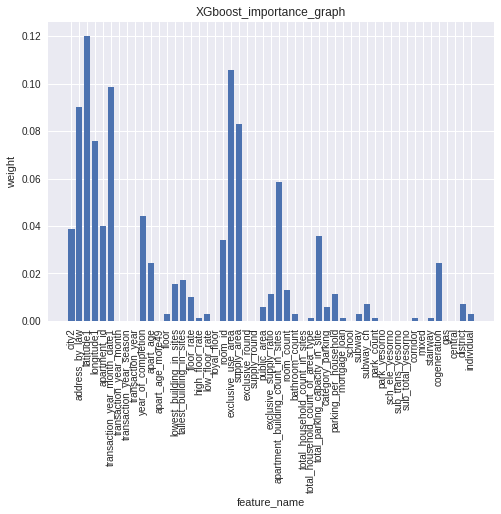
XGBoost의 장점은 Gradient Boosting의 기법 중 하나이기 때문에 병렬처리를 사용합니다. 따라서 학습과 분류가 빠르도 유연성이 좋습니다. 또한 다양한 옵션이 존재하여 모델을 합습시키는 것에 최적화를 시키며 평가를 할 수 있습니다. Scaling이 필요없고 Overfitting도 조금은 방지해 주기 때문에 초보자들이 사용하기 쉽습니다. 그리고 다른 알고리즘과 연계 활용성이 좋아 XGBoost 분류기 결론부 아래에 다른 알고리즘을 붙여서 앙상블 형태로 학습이 가능합니다.

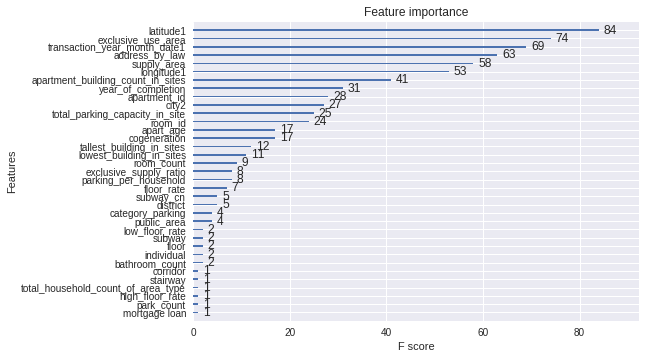
모델 훈련과 예측에 필요한 파라미터들을 다음과 같이 설정하였습니다.[표6]

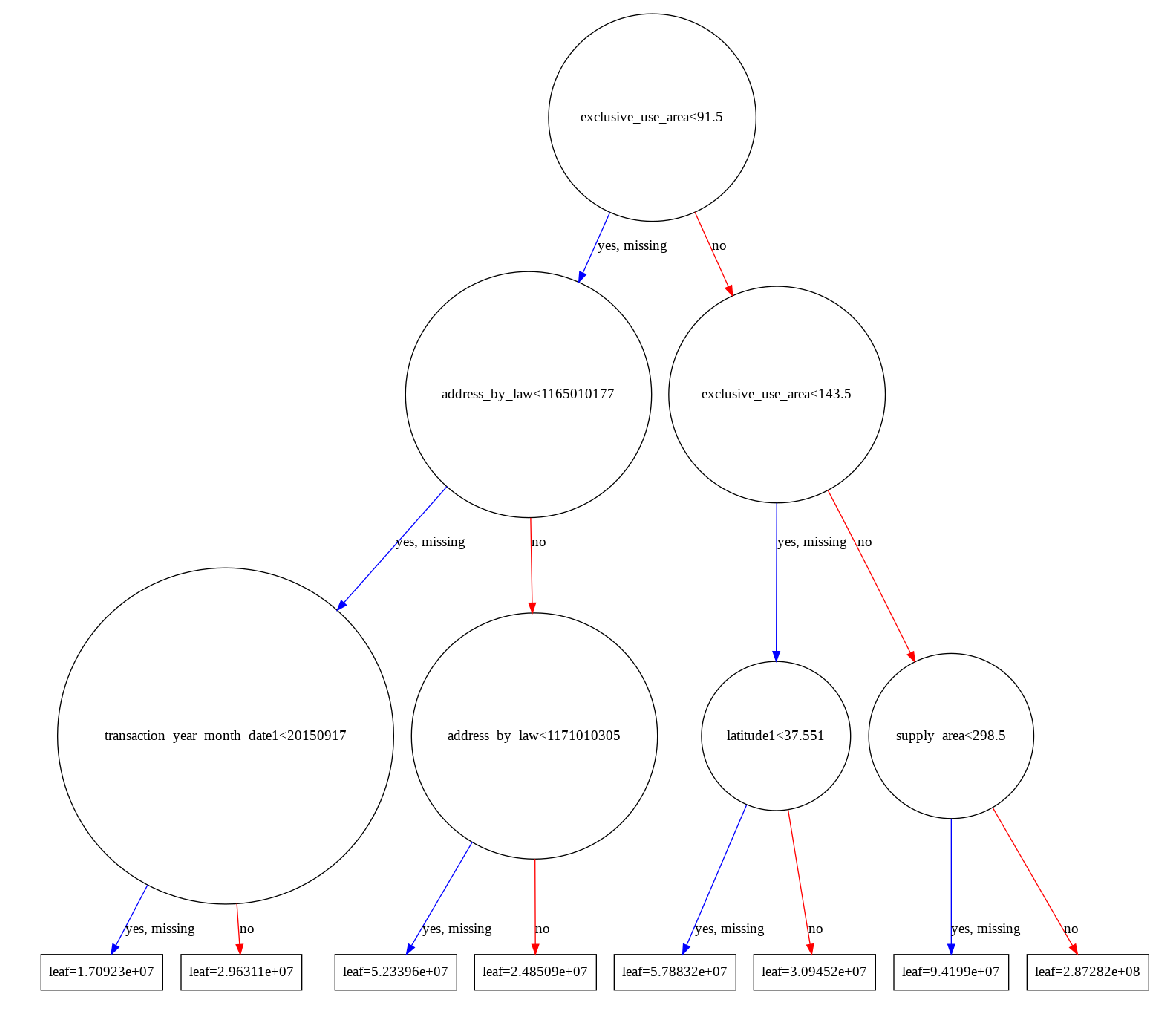
[표6. XGBoost 파라미터]

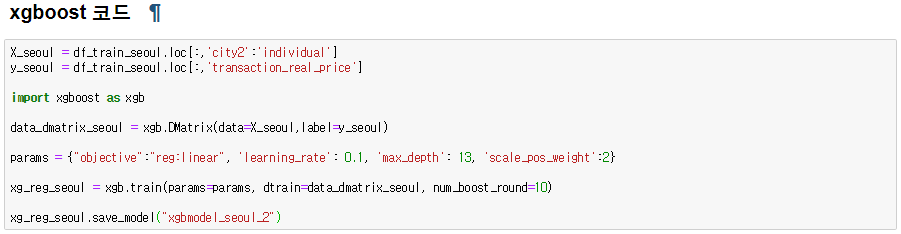
|  |  |  |
| --- | --- | --- |
| 순서 | Parameter | 비고 |
| 1 | max\_depth | 한 개의 Tree의 최대 깊이를 의미. |
| 2 | eval\_metric | Loss function(손실함수), 오차 측정 기준을 의미한다. ‘RMSE’ 설정. |
| 3 | num\_rounds | Boosting의 Round를 결정한다. . |
| 4 | gamma | Node가 Split 되기 위해 손실함수의 값이 감소하는 최소 수치. gamma값이 높을 수록 알고리즘은 보수적으로 변화한다. |

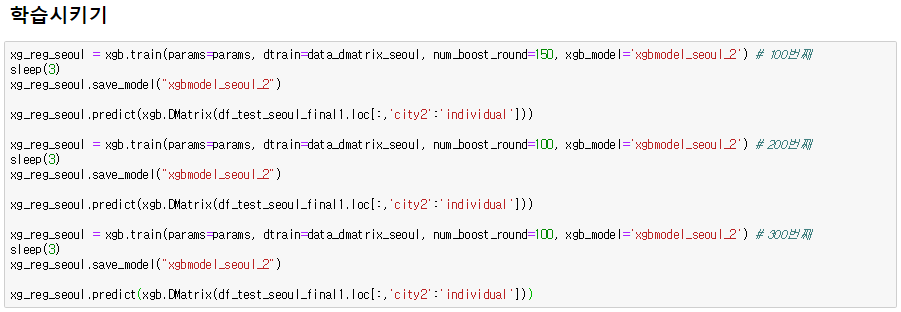
한편, feature Importance를 통해 모델이 어떤 feature에 얼마나 의존하고 있는지를 알 수 있습니다. 아래의 그래프를 보면 변수의 중요도에서 아파트의 위도와 면적, 거래날짜, 법정동 등이 아파트 실 거래가에 관한 가장 중요한 변수라고 예측하고 있습니다. 이것을 바탕으로 변수들의 가중치 그래프도 넣어봤습니다.

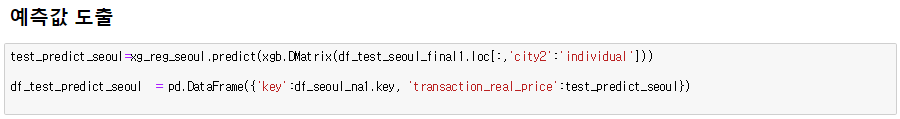












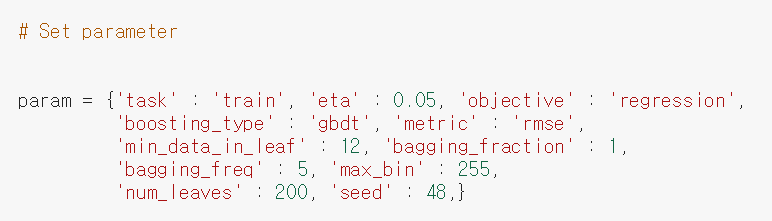
3.5.3 LightGBM

그라디언트 부스팅 결정 트리(GBDT, 혹은 GBM)는 효율성, 정확도, 해석 가능성이 높아 널리 사용되는 기계학습 알고리즘입니다. GBDT는 멀티 클래스 분류, 클릭 예측, 순위 학습 같이 다양한 작업에서 높은 성능을 보여줍니다. 하지만 빅데이터의 등장으로 정확도와 효율성간의 Trade off라는 새로운 숙제에 직면하게 됩니다.

그라디언트 부스팅 알고리즘(GBM)에는 XGBoost와 pGBRT와 같이 효율적으로 구현된 기술들이 존재하는데, 기존 방식에서는 각 변수마다 가능한 모든 분할점에 대해 정보 획득을 평가하기 위해 데이터 개체 모두를 점검합니다. 이 때의 계산 복잡도는 변수 개수와 개체 수에 비례하고, 빅데이터 처리 시 해당 구현은 많은 자원을 요구하게 되고, 속도 또한 느려지는 문제가 있었습니다. 이에 기울기 기반 단측 표본추출(GOSS)과 배타적 변수 묶음(EFB)이라는 두 가지 기술을 통해 기존 문제점을 해결하고자 하였다. 즉, 기존 GBDT에 GOSS와 EFB를 적용한 것을 LightGBM이라 부르게 되었습니다.

먼저 기울기 기반 단측 표본추출에서는 먼저 기울기가 작은 개체에 대해 무작위 표본 추출을 수행합니다. 다음으로 정보 획득을 계산할 때는 기울기가 작은 표본 데이터를 특정 상수만큼 증폭시키면 기존의 데이터 분포를 많이 변경하지 않고도 훈련이 덜 된 개체에 초점을 보다 잘 맞추게 됩니다. 데이터 전체 개체를 사용하지 않고도 일반화 성능을 달성, 향상할 수 있게 되는 것입니다.

다음으로 변수 개수를 효과적으로 줄이는 새로운 방법으로 배타적 변수 묶음(EFB)을 제시되었습니다. 변수 탐색 알고리즘을 통해 개별 변수들의 히스토그램(히스토그램 변수)과 동일한 것을 변수 묶음을 통해 만들어내고, 이 때의 히스토그램 생성 복잡도는 기존보다 훨씬 감소하게 된다. 따라서 정확도를 저하시키지 않으면서 GBDT 훈련 속도를 향상할 수 있습니다.[[6]](#footnote-6)



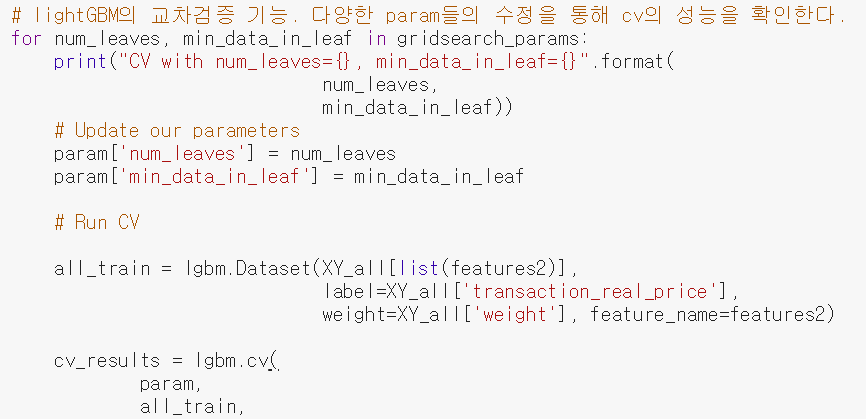
모델 훈련과 예측에 필요한 파라미터들을 다음과 같이 설정하였습니다.[표7]

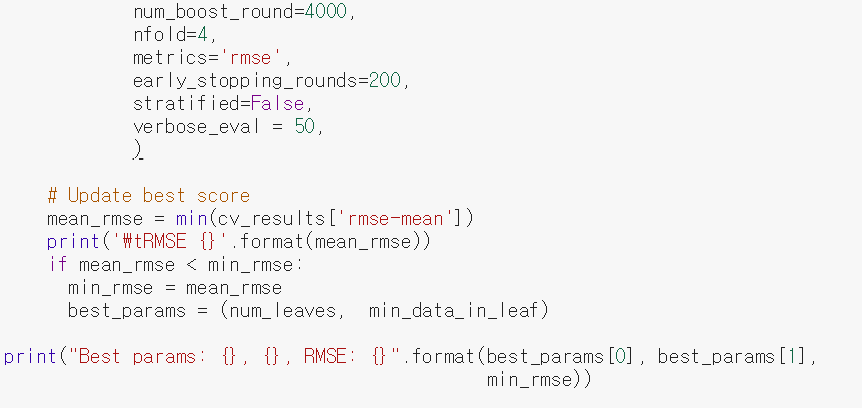
[표7. lightGBM Parameter]

|  |  |  |
| --- | --- | --- |
| 순서 | Parameter | 비고 |
| 1 | eta | learning rate; 예측기에 적용되는 학습률 가중치. |
| 2 | metric | Loss function(손실함수), 오차 측정 기준을 의미한다. |
| 3 | min\_data\_in\_leaf | 잎사귀 노드 한 개에 들어갈 수 있는 데이터의 개수. |
| 4 | bagging\_freq | bagging의 빈도. n회의 round가 진행된 뒤 1번의 bagging을 진행. |
| 5 | num\_leaves | 한 개의 트리에서 잎사귀 개수. |
| 6 | max\_bin | 변수 값들이 담긴 bins의 최대 수. |

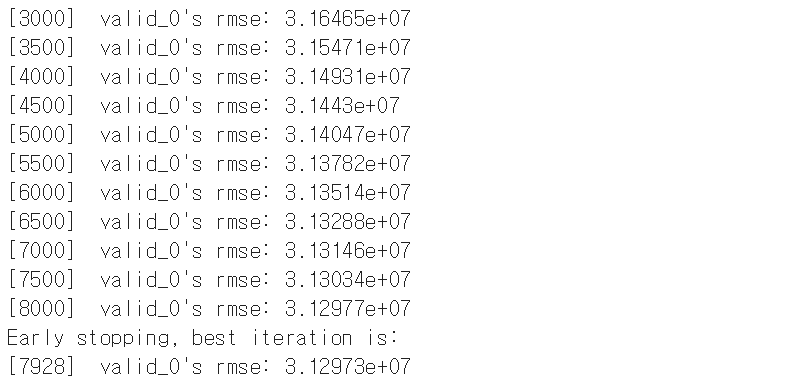
해당 parameter값들은 다음의 Cross Validation을 통해 얻어낸 최적의 값에 해당합니다.











**4. 프로젝트 결과 및 개선사항**

4.1 프로젝트 결과 및 개선사항

저희 듀란듀란 팀은 비 재무/금융 데이터를 활용하여 서울지역 아파트 거래가격을 분석, 예측하는 프로젝트를 진행했습니다. 훈련 데이터에서 사용한 변수는 공급면적과 층수, 준공 년도와 같은 아파트 자체의 정보 변수와 아파트 주변 위치정보 데이터(학교, 지하철, 공원) 등을 종합하여 예측 모델을 구성하였습니다.

최적의 예측 모델을 구성하기 위해서 다양한 분석, 예측 알고리즘을 사용하였습니다. OLS, Random Forest와 같은 기본적인 머신러닝 알고리즘부터, Keras로 진행한 딥러닝 알고리즘, 그리고 예측력과 속도에 장점이 있는 그라디언트 부스팅 기반의 XGBoost와 lightGBM까지 분석 모델링을 진행하였습니다. 그 결과 그라디언트 부스팅 기반의 XGBoost에서 가장 좋은 예측률(낮은 오차)을 보였으며, 테스트 데이터로 확인한 최종 오차 값은 62,579,859원에 해당하였습니다.

이번 프로젝트의 목표는 크게 두 가지였습니다.

먼저, 아파트 단지 정보, 매매 정보 등 다양한 비금융 데이터를 기반으로 보다 정밀하게 아파트 실 거래가를 예측하는 모델을 만드는 것입니다. 기존의 금융데이터와 시계열 데이터를 제외하고도 분석 모델을 만들 수 있다면 그 자체로 성공적이라고 생각했기 때문입니다. 프로젝트 진행 결과 6천만원의 오차결과값이 나왔지만, 시계열 데이터와 금융 정보 등을 포함하지 않고 평가모델을 만들고, 용인할 수 있는 오차범위 안에 위치한 것은 큰 수확이라고 생각합니다.

다음으로 모델의 예측 오차를 줄여 단순히 분석 모델을 구현하는 것에서 그치는 것이 아니라 기존 가치 산정방법들을 대체하거나 보완할 수 있도록 성능을 극대화 시키는 것이었습니다. 이 부분이 팀 차원에서 아쉬운 부분으로 남았습니다. 시간적 여유가 더 있었다면, Feature Engineeing을 좀 더 세부적으로 진행하고 모델의 Fine Tuning을 통해 최적의 모델 성능을 나타내는 파라미터를 더 찾을 수 있지 않았을까 생각합니다. 향후 시간적 여유가 확보된다면 모델의 성능을 좀 더 끌어올릴 수 있을 것이라 생각합니다.

4.2 기대 효과

4.2.1 개인

한국 사회에서 부동산은 국민의 삶과 매우 밀접한 연관을 지닌 요소입니다. 최근에는 추세가 변화하고 있다지만, 예전부터 부동산은 개인의 자산 목록에서 가장 중요한 요소였습니다. 대한민국 국민 중 아직 48.1%는 아파트에서 거주하고 있으며, 생활에 있어서 매우 밀접한 요인임을 부정할 수 없습니다.

그러나 이러한 중요성에 비해 부동산 시장에 대한 정보는 접근이 상당히 제한되어 있습니다. 중개인을 통한 거래가 보편화된 현재 시장에서 특이하게도 부동산 중개인마다 같은 요건의 아파트에 있어서도 제시하는 가격이 차이가 나는 경우가 있습니다. 개인의 입장에서는 관련 정보에 대한 접근의 제한이나 획득 가능한 정보량의 부족으로 더 나은 선택지를 포기하거나 아예 인지조차 못하는 상황에 처할 가능성이 생깁니다. 정보의 비대칭성과 불 투명성이 존재하는 것입니다. 이번 프로젝트를 통해서 이러한 문제점들을 해결할 단초를 제시하는 기대 효과를 기대할 수 있습니다.

최근 부동산과 공공 데이터를 연결하려는 노력이 증가하고 있는 추세입니다. 실 거래가를 중개인을 통해서만 얻을 수 있는 것이 아니라 관련 변수들에 대한 정보를 얻는다면 그것을 기반으로 개인이 실 거래가를 예측함으로써 거래의 기준점으로 활용하여 피해를 줄일 수 있습니다. 혹은 잘못된 정보를 가지고 사기를 치려는 중개인의 의도를 피해갈 수도 있을 것입니다.

4.2.2. 기업

4.2.2.1. 부동산 중개정보 서비스 기업

부동산 실 거래가 예측 시스템을 통해서 직방, 다방, 다음 부동산, 네이버 부동산 등 중개사와 구매자를 연결시켜주는 서비스를 시행하고 있는 기업들이 얻는 장점이 있습니다. 개인의 이점에서 언급한 정보의 비대칭성과 불 투명성을 해결할 경우에 부동산정보 서비스 시장의 신뢰도를 높일 수 있다는 점입니다. 신뢰도 회복 및 상승을 통해서 시장 자체의 성장을 도모할 수도 있으며, 경쟁사에 비해서 높은 신뢰도를 확보함으로써 경쟁력을 높일 수 있다는 장점을 얻을 수 있습니다.

다만, 앞서의 장점을 위해서는 선행되어야 하는 점이 있습니다. 신뢰도 높은 예측 프로그램을 구축하는 것입니다. 여러 부동산 관련 사이트들에서 관련된 자료들을 확보하고 제공하고 있습니다. 그러나 최근 매물에 대한 정보는 즉각적으로 업데이트 되기에는 제한이 있는 상황입니다. 그러한 상황에서 최근 매물에 대한 정확도 높은 실 거래가 예측이 가능한 프로그램이 있다면, 해당 정보에 대한 신뢰를 통해서 거래 활성화 및 거래량 증가의 결과를 얻을 수 있을 것입니다.

4.2.2.2. P2P 금융 기업

부동산을 기반으로 한 금융 상품들은 매우 다양합니다. 아파트의 경우에도 2가지의 대출 상황이 있습니다. 첫 번째로 아파트를 구매하고자 하는 사람들이 구매 금액을 한번에 지불할 능력이 안될 경우에 구매할 아파트를 담보로 대출을 하는 경우입니다. 두 번째로는 상환한 아파트를 담보로 하여 새로운 대출을 받는 경우입니다. 이외에도 더 많은 상황들에 따른 파생상품들을 생성할 수 있습니다.

그러나 상품이 다양하더라도 상품에 대한 수요가 충분하지 못하다면 의미가 없을 것입니다. 신용 등급이나 경제적 여건 외의 다른 요소로 인해 기존 금융권들로부터 담보 대출이 불가능한 고객들이 있습니다. 그러한 고객들 역시도 잠재적인 수요로 볼 수 있습니다. 다만, 위험도가 높은 수요에 대한 불안심리로 인해 기업들 역시도 마음껏 대출상품을 제공할 수 없습니다. 이러한 기존 금융권들이 놓치고 있는 수요에 대해서 P2P 금융 기업들은 기회를 얻을 수 있을 것입니다.

기존 금융권들만 사용 가능한 재무, 금융 정보들을 제외한 비 금융 정보들을 통해서 아파트나 그 외 부동산에 대한 가치 예측이 가능하다면, 이는 새로운 상품 기획 및 활용에 있어서 획기적인 기회일 것입니다. 고객의 입장에서는 충분히 상환 능력이 있음에도 잘못된 기준으로 기존 금융권으로부터 받을 수 없었던 대출 기회를 얻을 수 있으며, P2P 기업들의 경우에는 금융 정보의 부재로 인해 접근할 수 없었던 기존 대출 시장과 수요 고객들에 대한 기회를 얻을 수 있습니다. 새로운 시장 개척 및 고객 확보를 통한 이득을 얻을 수 있을 것입니다.

**5. 분석 후기**

|  |
| --- |
|  |

|  |  |
| --- | --- |
| 성명 | 후기 |
| 유선우 | 과정을 시작할 때는 컴퓨터 언어는 하나도 모르고 관련 전공수업은 들어본 적도 없는 상황이었습니다. 빅데이터라는 분야에 대한 전공 지식 및 경험이 일체 없었던 저에겐 모든 것이 도전이었고, 두려움이었습니다. 불안과 걱정 속에 시작했던 과정이었지만, 지금 이렇게 6개월의 시간을 보내고 끝을 마주한 상황에선 만족과 보람을 느끼고 있습니다.  그렇게 변화된 이유 중 가장 큰 것은 같은 교실에서 함께 수업을 받은 친구들이었습니다. 그 중에서도 지금의 팀원들이 정말 큰 힘과 의지가 되어 주었습니다. 강사님을 비롯해 멘토님들이 공통적으로 하신 말씀 가운데 ‘현재 같이 공부하는 친구들이 미래 인맥이며, 과정이 끝난 이후 사람들만 남겨도 성공한 것이다.’였습니다. 모르는 것을 서로 물으며 배워가고, 힘든 일은 도와주는 협동을 배우고 함께하는 팀 프로젝트의 매력을 느낄 수 있었던 것은 재능 넘치는 동료들 덕분이었습니다. 과정의 종료가 아쉬워질 만큼 함께한 시간이 즐겁고 보람찼습니다. 이후로도 이러한 인연을 잘 유지해 각자 어느 위치에 서던지 상부상조할 수 있는 모임이 되었으면 좋겠습니다.  그러한 만족과 보람 외에도 아쉬움과 부족함을 깨닫는 계기도 되었습니다. R과 Python을 배웠지만, 데이터 분석이라는 영역에서 능력을 발휘하기 위해서는 더욱 많은 지식과 관련 경험이 필요하다는 것을 깨닫게 되었습니다. 3번의 미니 프로젝트와 본 프로젝트의 경험을 쌓았지만, 매번 느끼는 것은 아직 부족하다는 점과 더 많은 경험이 필요하다는 자각이었습니다. 단순히 컴퓨터 프로그램을 활용하는 능력 외에도 분석이라는 점에 있어서 통계적인 지식과 문제를 바라보는 다양한 시각을 제시할 수 있는 감각이 필요하다는 것을 깨달았습니다. 같은 문제를 바라보고서도 다양한 문제 인식과 그에 따른 다양한 해결 방식이 나올 수 있다는 점을 생각하며 항상 사고하고 고민하는 자세에 대해 깨우치게 되었습니다  6개월이라는 시간 역시 바라보는 관점에 따라 길거나 짧은 기간입니다. 두렵고 낯선 첫 순간에는 길고도 먼 과정이었지만, 함께 배우고 정진해 나간 이후 돌아본 그 기간은 너무도 짧고 아쉬운 순간처럼 느껴집니다. 과정 이후 관련 현업에 활동하거나 새로운 영역을 도전하는 선택을 앞둔 지금 다시 또 길고 먼 미래를 바라보게 되었습니다. 이후 어느 순간엔가 돌아봤을 때, 후회 없이 노력하고 정진해 나간 시간들이 너무나도 아쉽고 즐거워 짧게 느껴질 수 있도록 계속 노력할 것입니다. |
| 장정호 | 지난 6개월 간의 프로그래밍과 데이터에 대한 이해, 분석에 대한 기초를 배우면서 처음에는 개념적인 이해도 어려웠던 기억이 납니다. 중간 중간 어려움을 겪은 시간들이 저를 힘들게 했지만, 도리어 그러한 고난이 있어서인지 지금은 조금씩 재미도 붙여나가고 있고, 작지만 자신감도 붙고 있습니다.  프로젝트를 진행한 두 달여의 시간은 그 동안 배운 것들을 녹여내는 새로운 장이었습니다. 허주용 멘토님과 김진수 강사님의 지도 아래 저의 미진한 부분과 개선할 부분을 해결해나가면서 배운 것들을 복습하고, 새로운 것들을 담아 프로젝트에 반영해왔습니다.  프로젝트가 거진 마무리 되었지만 조금의 시간이 더 있었더라면 더욱 알찬 내용으로 채울 수 있을 것 같아 아쉬움이 많이 남습니다. 그러나 지금의 결과물 또한 저희 팀이 하나가 되어 이루어낸 것이라 이 자체로도 자랑스럽습니다.  비록 교육과정은 마무리 되었지만, 이를 통해 데이터 분석 직무로 사회에 진출할 첫 걸음을 떼었다고 생각합니다. 지금보다 더 분석적인 사고와 데이터를 바라보는 시야를 넓히고, 그것을 풀어낼 프로그래밍 실력을 더 키우고자 합니다.  마지막으로 6개월을 함께한 같은 반 친구들과 무엇보다도 부족한 저를 많이 도와준 우리 프로젝트 팀의 친구들 덕분에 행복했습니다. 그들에게 감사하다는 말을 전하고 싶습니다. |
| 민유진 | 3~4 개의 미니 프로젝트, 공모전, Python 교육 등으로 많이 준비가 되었을 것이라고 생각했지만 큰 오해였습니다. 이번 프로젝트의 큰 주제인 회귀에도 수많은 기법들이 있었고 그것들을 이해하는 것도, 구현하는 코딩 실력에도 많은 부족함이 있음을 느꼈습니다. 하지만 이번 프로젝트로 XGBoost라는 핫한 Gradient Boosting 기법을 공부할 수 있게 되어서 좋았습니다. 교육과정이 끝나면 회귀, 분류 모델에 대한 공부를 더 해보고 싶고 통계공부도 더 할 것입니다.  멘토님께서 멘토링 첫날부터 말씀하셨던 변수를 만들지 못한 아쉬움이 큽니다. 당장 지금이 아니더라도 프로그래밍 실력을 더 키워서 꼭 구현해보고 싶습니다. 현업에서 데이터분석을 하시면서 생기는 현실적인 문제나 취업을 하기 위한 자소서, 면접에 대한 조언들도 많이 해주셨습니다. 또한 맛있는 것들도 많이 사주셔서 좋은 곳에 취업해서 맛있는 것을 대접해드리고 싶습니다.  또한 코딩에 대한 어려움과 친하지 않은 사람과 토론하는 것이 쉽지 않았는데 좋은 인연으로 듀란듀란을 만나게 되어 좋았습니다. 부족한 점이 많았지만 항상 웃으면서 좋게 봐주고 재밌는 분위기로 인해 스트레스 없이 많은 것을 배우면서 즐겁게 프로젝트에 임한 것 같습니다.  코딩에 대해 자신이 없었는데 교육과정을 통해 자신감과 흥미가 생겼습니다. 또한 통계나 여러 회귀 기법들, 다른 프로그래밍 언어를 배워야겠다는 생각이 들었고 다른 공모전이나 kaggle에도 참여해보고 싶은 욕심이 생겼습니다. 데이터 분석에 대한 꿈이 생겼고 그 꿈을 위해 열심히 살 것입니다!!! |
| 함윤선 | 프로젝트를 하면서, 효율적인 데이터 분석을 하기 위해서는 해당 분야의 도메인 지식과 수리통계 지식이 기반되어야 함을 느꼈습니다. ‘파생변수 몇 개 만들고, 모델 최신 꺼 뚝딱 거리면 한 30위권 안에는 들 수 있겠지’ 라는 생각을 가지고 초반에는 접근했습니다. 그러나 파생변수 도출하기는 쉽지 않았습니다. 부동산 지식과 수리통계 지식이 전무하니, 어떻게 파생변수를 만들어야 하는지 감을 못 잡았으니까요. 또한, 외부데이터를 활용해 추가 변수를 만들어야 함에도, 어떤 외부 데이터를 찾아야 할지 잘 몰랐습니다.  그러나 허주용 멘토님 데이터분석 및 부동산 관련 지식을 아낌없이 친절히 열심히 멘토링을 해주셔서 보다 수월하게 진행 할 수 있었습니다. 물론, 팀원들의 공도 컸습니다. 팀원 민유진님의 수리통계적 지식 및 코딩 실력 , 팀장 유선우님의 꼼꼼한 정리와 부동산 지인 정보 그리고 팀원 장정호님의 모델 이론적 지식 및 정보 공유 없이, 저 혼자 이 프로젝트를 진행했었다면… 분명히 망했을 겁니다.  정말 좋은 멘토와 동료를 만나, 2달동안 즐겁게 프로젝트를 진행할 수 있었습니다. 이 자리를 빌어 모두에게 감사를 표하고 싶습니다. |

1. James Max Kanter(2015), Deep feature synthesis: Towards automating data science endeavors, *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA),* 19-21 Oct. 2015 [↑](#footnote-ref-1)
2. 김미숙(2018), 다층신경망의 다양한 연결구조 제안 및 분석, *한국정보기술학회논문지 제 16권 제4호*, 2018.4, Pages 11-19 [↑](#footnote-ref-2)
3. Jurgen Schmidhuber(2014), Deep Learning in Neural Networks: An Overview, *Neural Networks(Volume 61)*, January 2015, Pages 85-117 [↑](#footnote-ref-3)
4. Choi, H. and Min, Y. (2015).Introduction to deep learning. *Korea Information Processing Society Review*, 22, 7-21. [↑](#footnote-ref-4)
5. Lee, W. and Chun, H. (2016). A deep learning analysis of the Chinese Yuan’s volatility in the onshore and offshore markets. *Journal of the Korean Data & Information Science Society*, 27, 327-335. [↑](#footnote-ref-5)
6. Guolin Ke, Qi Meng(2017), *LightGBM: A Highly Efficient Gradient Boosting Decision Tree,* [Advances in Neural Information Processing Systems 30 (NIPS 2017](https://papers.nips.cc/book/advances-in-neural-information-processing-systems-30-2017)), 2017 [↑](#footnote-ref-6)