

PROJECT

Football Shots 데이터 기반 분류 모델 분석 보고서

For Orange3

CONTENT

01 | 프로젝트 개요

02 | 데이터셋 소개

03 | 슛 위치에 따른
스� 타입 분류 모델

04 | 슛 위치(positionX, Y)에 따른
골 분류 모델

05 | 결론

01. 프로젝트 개요

프로젝트 배경

경기 중 발생한 슈트(Shot) 데이터를 분석하여, 위치·행동·상황 등 경기 맥락(Context)이 슈트의 유형(Head/Foot/OtherBodyPart) 및 결과(Goal 여부)에 어떤 영향을 주는지를 탐구함

문제 정의

- 경기 내에서 발생하는 수천 건의 슈트를 단순히 “슈트”로만 구분하면 전술적 의도나 공격 패턴을 해석하기 어려움
 - 슈트 타입 분류 (Classification) → Head/Foot/Other 구분
 - 골 성공 예측 (Prediction) → positionX, Y를 기반으로 골 여부 판단

목표

- 경기 중 슈트 위치·행동·상황 변수를 이용해 슈트의 특성(타입)을 자동 분류
- 슈트 발생 위치를 중심으로 골 성공 확률을 예측하여, 선수 및 팀의 슈팅 효율성 분석 지표로 활용 가능성 탐색

데이터 출처

Kaggle Football Database 유럽 5대 리그 경기 데이터

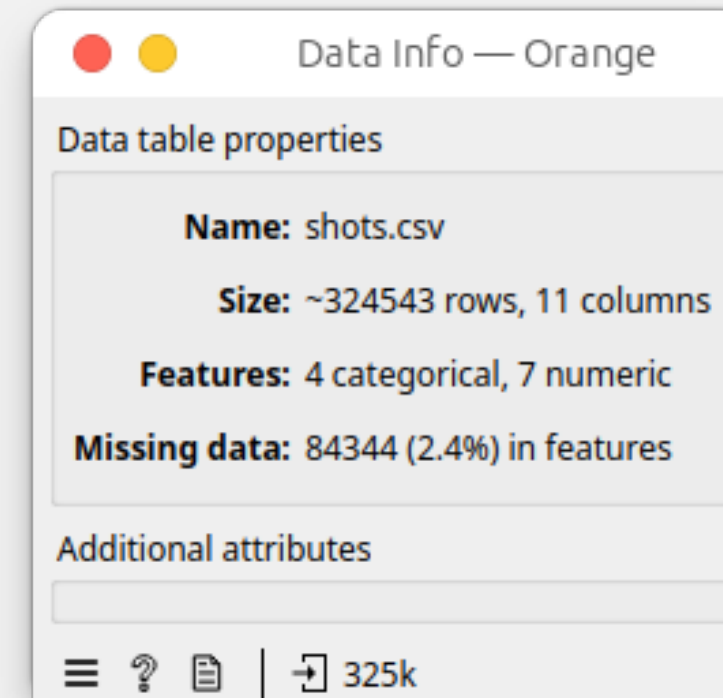


<https://www.kaggle.com/datasets/technika148/football-database?select=shots.csv>

02. 데이터셋 소개

데이터 규모

- 약 324,543건, 11개 컬럼
- 구성:
 - 4개의 범주형(categorical) 변수
 - 7개의 수치형(numeric) 변수
- 결측치: 약 84,344건 (2.4%)

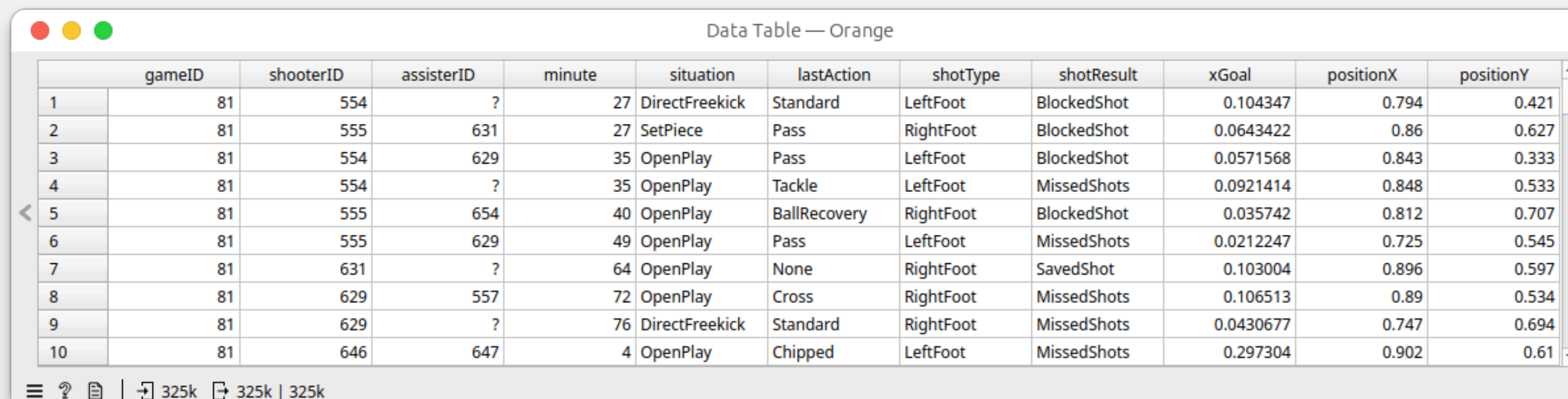


Data Info — Orange	
Data table properties	
Name:	shots.csv
Size:	~324543 rows, 11 columns
Features:	4 categorical, 7 numeric
Missing data:	84344 (2.4%) in features
Additional attributes	
≡ ? 325k	

주요 변수

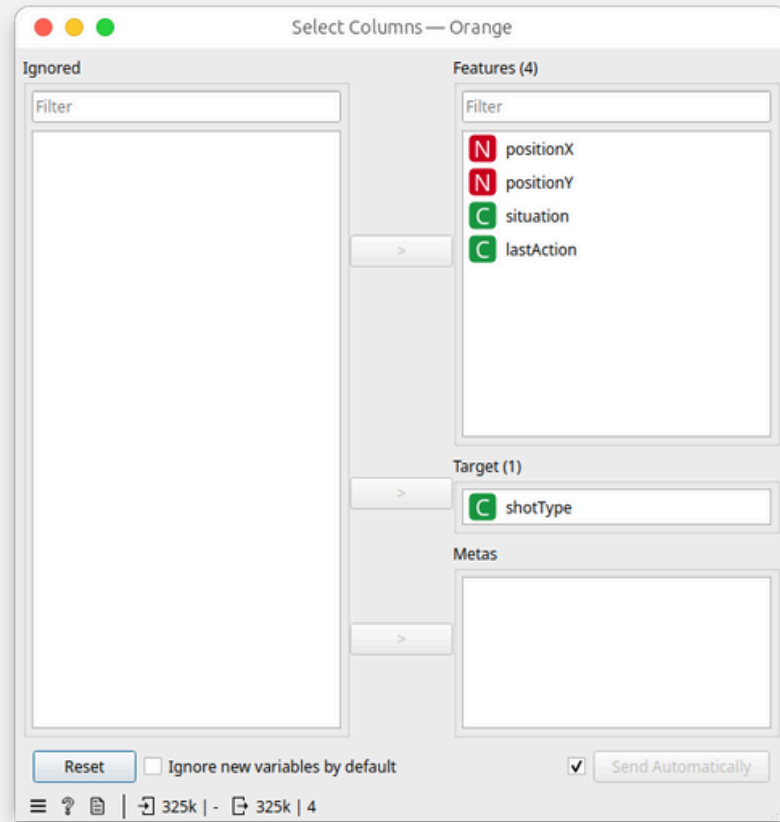
gameID: 경기 ID | shooterID: 슈팅 선수 ID | minute: 슈팅 발생 시점 | situation: 플레이 상황

lastAction: 직전 행동 | shotType: 슈팅 유형 | xGoal: 기대 득점 확률(0~1) | positionXY: 필드 내 슈팅 좌표 (0~1 스케일)

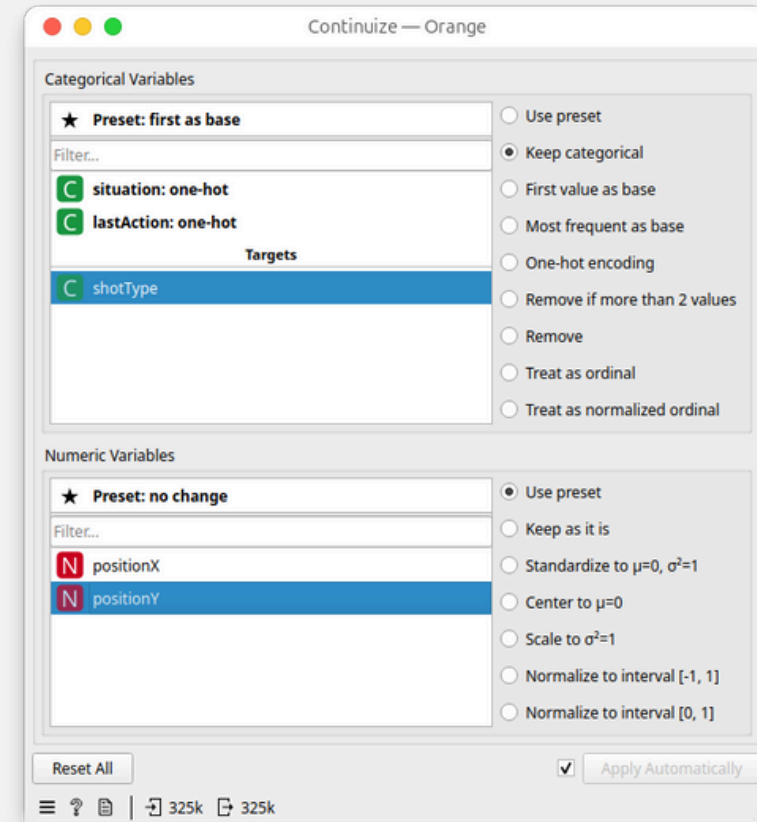


	gameID	shooterID	assisterID	minute	situation	lastAction	shotType	shotResult	xGoal	positionX	positionY
1	81	554	?	27	DirectFreekick	Standard	LeftFoot	BlockedShot	0.104347	0.794	0.421
2	81	555	631	27	SetPiece	Pass	RightFoot	BlockedShot	0.0643422	0.86	0.627
3	81	554	629	35	OpenPlay	Pass	LeftFoot	BlockedShot	0.0571568	0.843	0.333
4	81	554	?	35	OpenPlay	Tackle	LeftFoot	MissedShots	0.0921414	0.848	0.533
5	81	555	654	40	OpenPlay	BallRecovery	RightFoot	BlockedShot	0.035742	0.812	0.707
6	81	555	629	49	OpenPlay	Pass	LeftFoot	MissedShots	0.0212247	0.725	0.545
7	81	631	?	64	OpenPlay	None	RightFoot	SavedShot	0.103004	0.896	0.597
8	81	629	557	72	OpenPlay	Cross	RightFoot	MissedShots	0.106513	0.89	0.534
9	81	629	?	76	DirectFreekick	Standard	RightFoot	MissedShots	0.0430677	0.747	0.694
10	81	646	647	4	OpenPlay	Chipped	LeftFoot	MissedShots	0.297304	0.902	0.61

03. 슈트 위치에 따른 슈트 타입 분류 모델



1. 타겟 Columns 설정



2. 범주형 변수 인코딩

변환 데이터 — Orange

Info
324543 instances (no missing data)
48 features
Target with 4 values
No meta attributes.

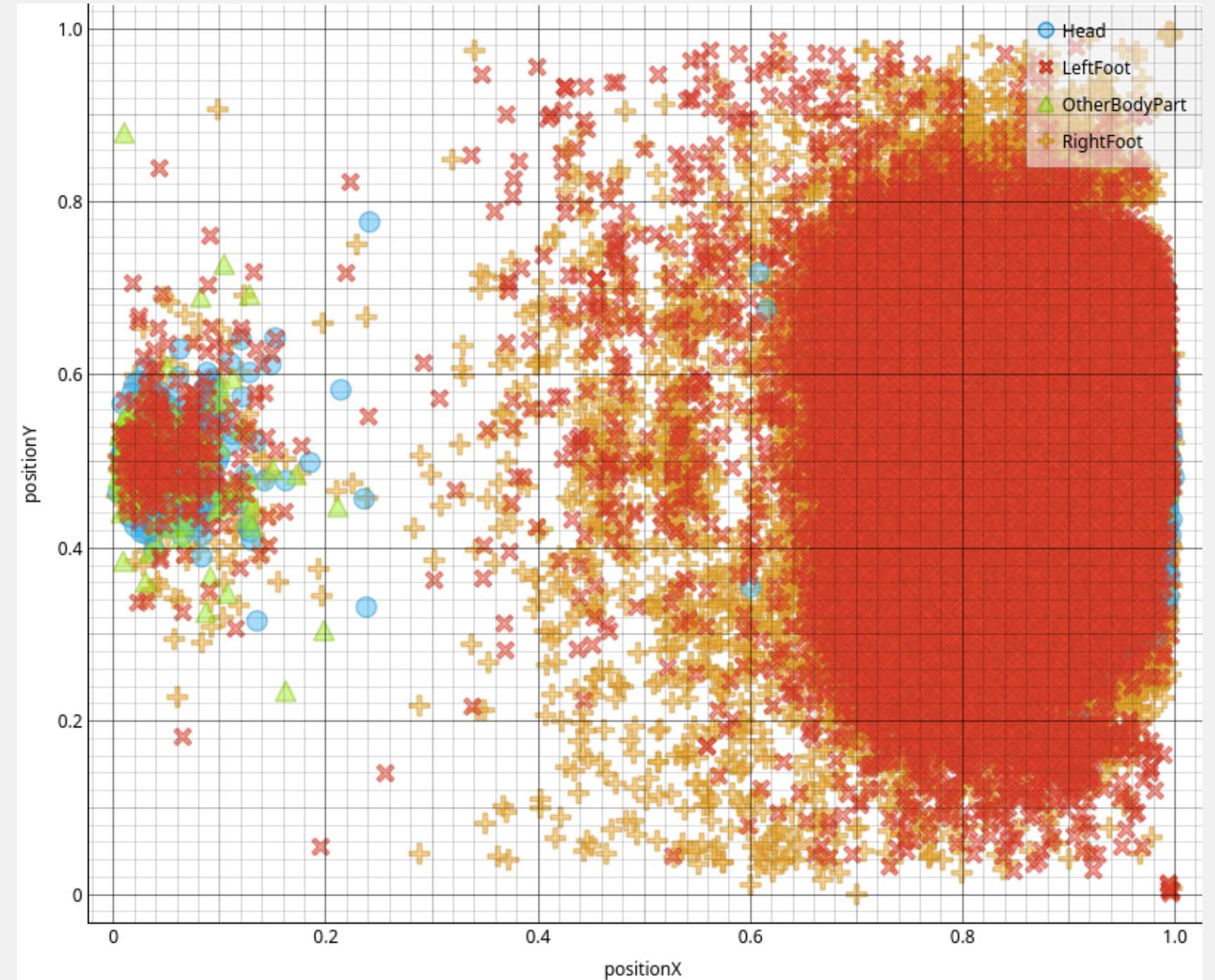
Variables
☒ Show variable labels (if present)
☐ Visualize numeric values
☒ Color by instance classes

Selection
☒ Select full rows

☒ Send Automatically

	shotType	positionX	positionY	ation=DirectFree	uation=FromCorr	ituation=OpenPla	situation=Penalty	ituation=SetPie	lastAction=Aerial	Action=BallRecov	stAction=BallTour	A
1	LeftFoot	0.794	0.421	1	0	0	0	0	0	0	0	0
2	RightFoot	0.86	0.627	0	0	0	0	1	0	0	0	0
3	LeftFoot	0.843	0.333	0	0	1	0	0	0	0	0	0
4	LeftFoot	0.848	0.533	0	0	1	0	0	0	0	0	0
5	RightFoot	0.812	0.707	0	0	1	0	0	0	1	0	0
6	LeftFoot	0.725	0.545	0	0	1	0	0	0	0	0	0
7	RightFoot	0.896	0.597	0	0	1	0	0	0	0	0	0
8	RightFoot	0.89	0.534	0	0	1	0	0	0	0	0	0
9	RightFoot	0.747	0.694	1	0	0	0	0	0	0	0	0
10	LeftFoot	0.902	0.61	0	0	1	0	0	0	0	0	0
11	RightFoot	0.879	0.247	0	0	1	0	0	0	0	0	0
12	Head	0.866	0.497	0	1	0	0	0	0	0	0	0
13	RightFoot	0.831	0.433	0	1	0	0	0	0	0	0	0
14	RightFoot	0.936	0.353	0	0	1	0	0	0	0	0	0
15	LeftFoot	0.095	0.499	0	0	1	0	0	0	0	0	0

전처리가 완료된 테이블



슈트 타입에 따른 위치 분포

03. 슈트 위치에 따른 슈트 타입 분류 모델

4. 학습 데이터 분할

- Random Sampling
 - Train 70%, Test 30%

5. 머신러닝 모델 구축

- Logistic Regression – 단순하고 해석력 높음, baseline 역할
- Random Forest – 복잡 패턴 포착, 정확도 향상
- AdaBoost – 오분류 보완, 안정성 강화

6. 결과

- 정확도(CA) 가 0.61~0.62 수준 → 모델이 4개 클래스를 구분하기 어려움
- F1 Score (0.50~0.60) 도 낮음 → 예측의 일관성(Precision/Recall) 모두 낮은 편
- MCC (0.32~0.39) → 분류 신뢰도 중간 이하

Model	AUC	CA	F1	Prec	Recall	MCC
Logistic Regression (1)	0.734	0.620	0.505	0.578	0.620	0.399
Random Forest (1)	0.750	0.613	0.606	0.602	0.613	0.360
AdaBoost (1)	0.708	0.592	0.588	0.586	0.592	0.325

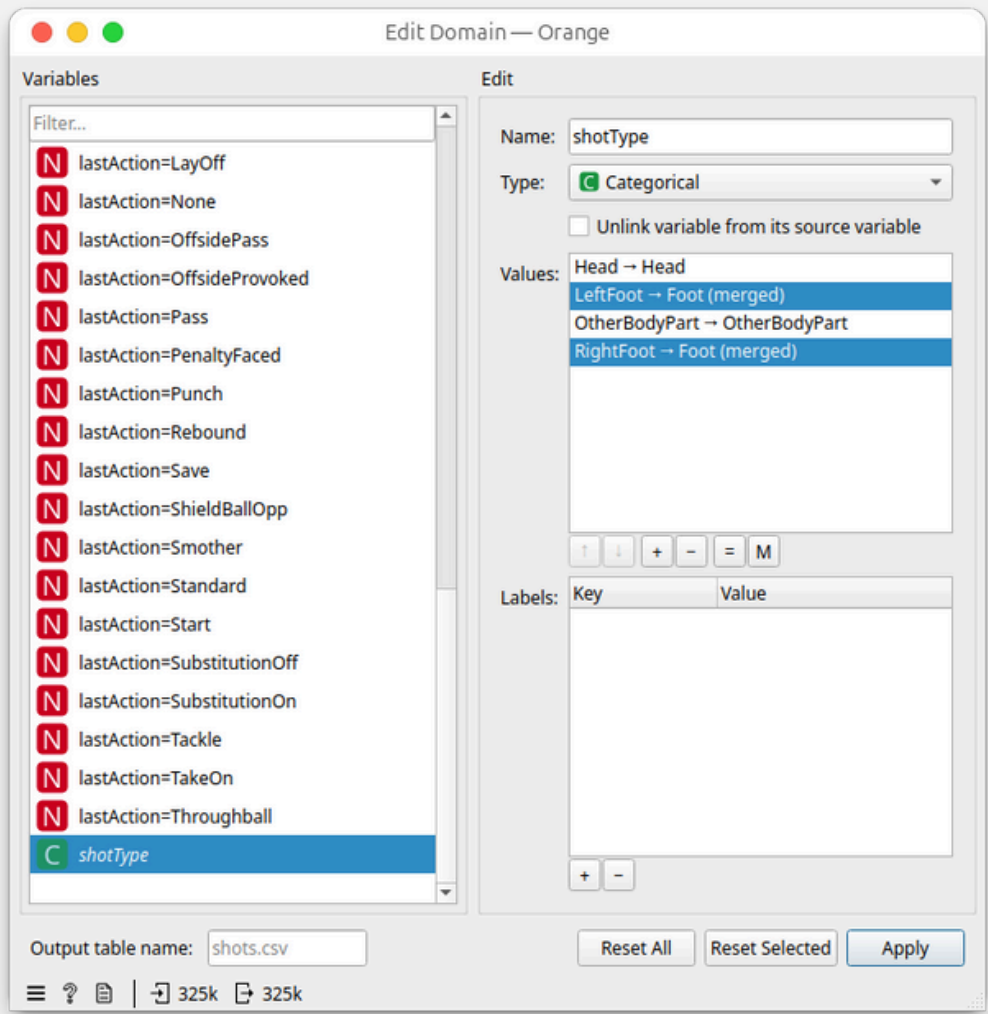
=> 모델이 전체적으로 패턴을 어느 정도는 인식하지만, 클래스 간 구분이 명확하지 않음

7. 정확도 저하 요인 분석

- 데이터 특성상 RightFoot/LeftFoot의 피쳐 패턴이 매우 비슷해 모델이 구분 어려움
- 4개 클래스를 동시에 구분해야 하므로 결정 경계 복잡성 ↑

8. 개선

- Edit Domain을 통해 shotType의 LeftFoot과 RightFoot을 한 클래스로 Merge



	shotType	positionX	positionY	action=DirectFree	uation=FromCorr
1	Foot	0.794	0.421	1	0
2	Foot	0.86	0.627	0	0
3	Foot	0.843	0.333	0	0
4	Foot	0.848	0.533	0	0
5	Foot	0.812	0.707	0	0

03. 슈트 위치에 따른 슈트 타입 분류 모델

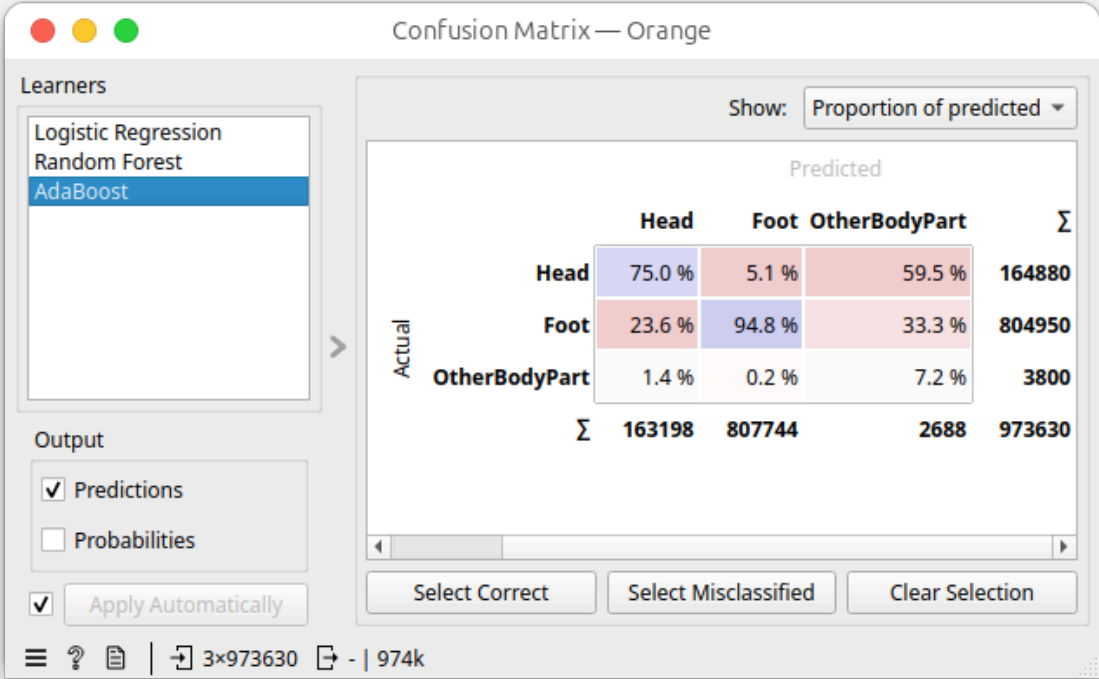
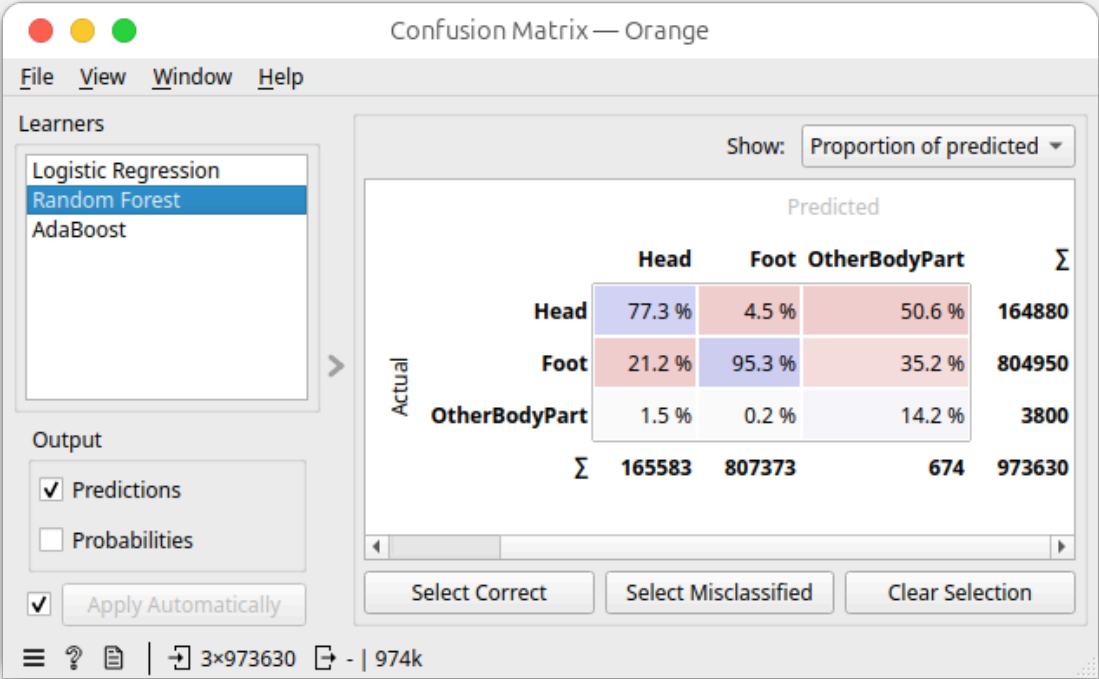
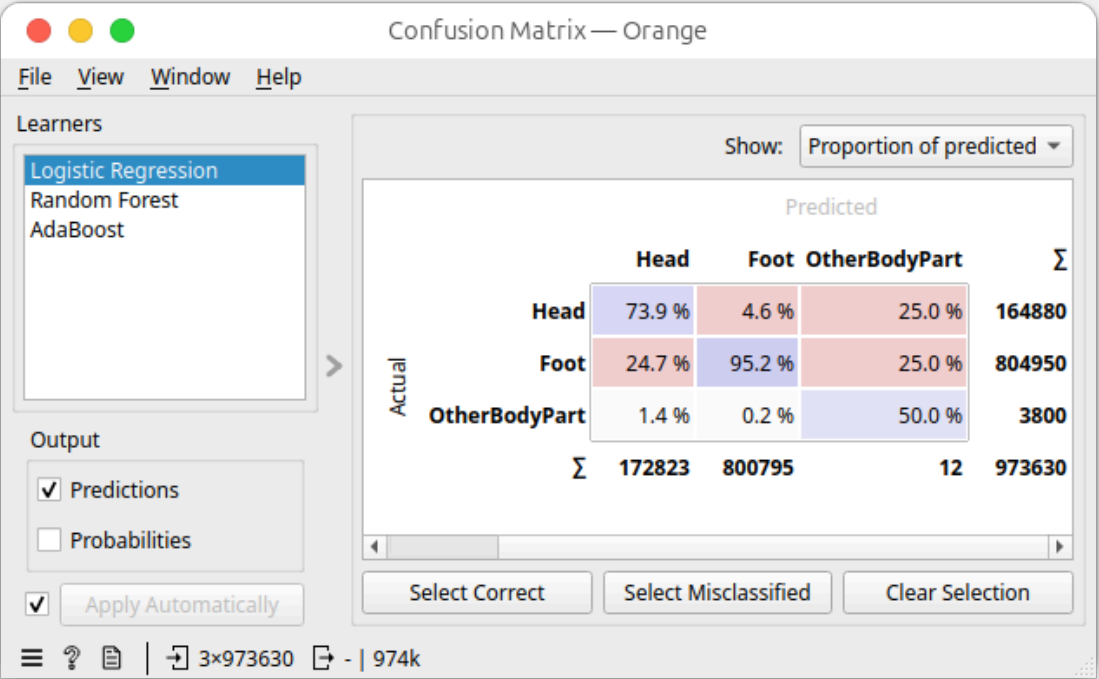
9-1. Score 결과

- 정확도(CA): 약 91~92%
- AUC (0.92~0.96): 클래스 간 분리도가 높음 → 모델이 각 슈트 타입을 잘 구분
- F1, Precision, Recall 모두 균형적 (0.91~0.92) → 오탐과 미탐 모두 적음
- MCC (0.70~0.73) → 전반적 분류 신뢰도도 높음

Model	AUC	CA	F1	Prec	Recall	MCC
Logistic Regression	0.959	0.914	0.913	0.914	0.914	0.704
Random Forest	0.955	0.922	0.921	0.920	0.922	0.728
AdaBoost	0.921	0.912	0.912	0.911	0.912	0.693

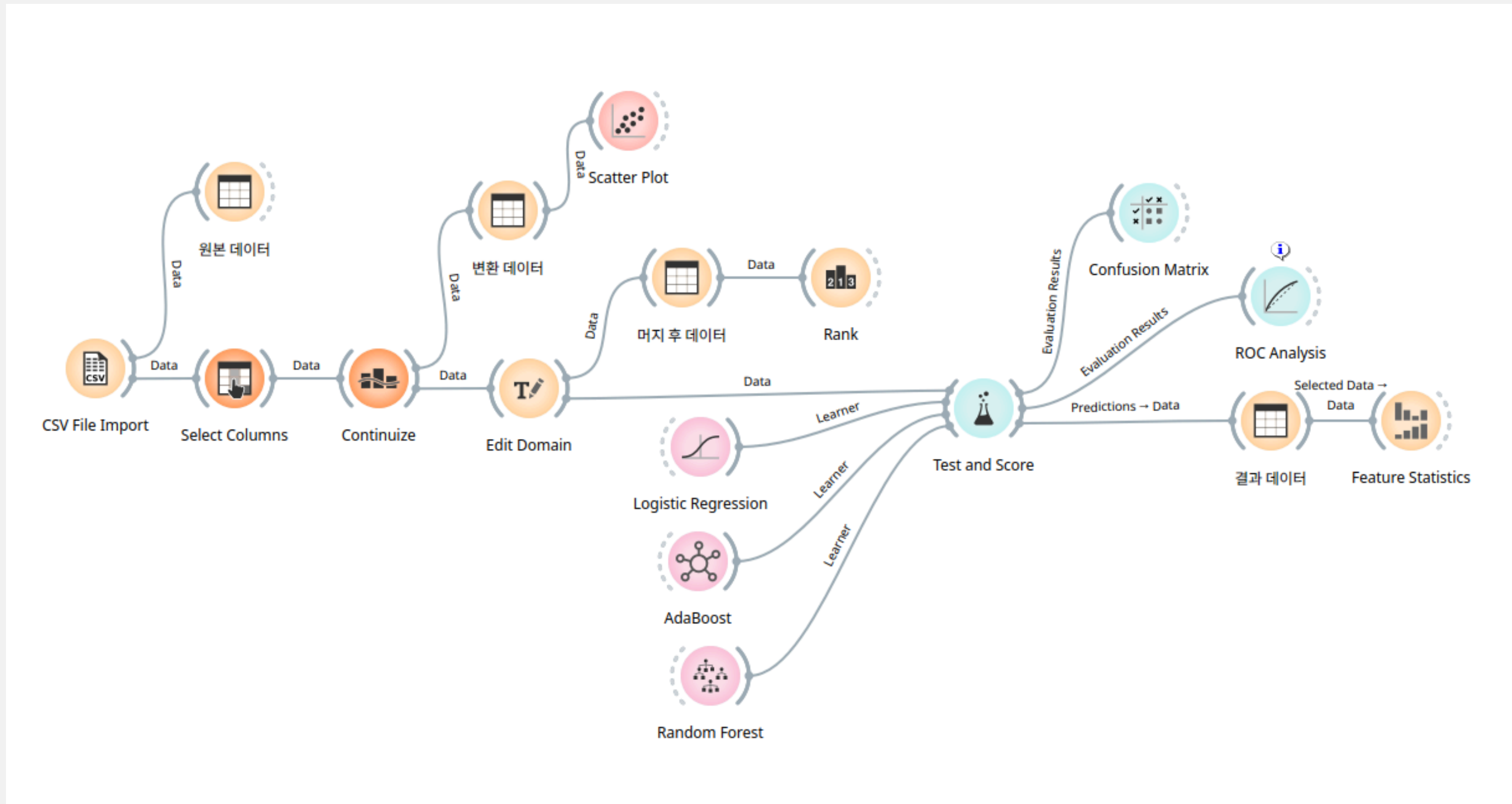
9-2. Confusion Matrix 결과

- Logistic Regression: 단순 선형 모델이라 Head ↔ Foot 간 경계가 뚜렷하지 않을 때 혼동이 발생했지만, 전체적으로 클래스 불균형에도 불구하고 꽤 안정적인 분류를 수행
- Random Forest: 비선형 관계(positionX, lastAction 등) 를 잘 포착하여 Head ↔ Foot 패턴을 가장 잘 구분함... 실제 경기 맥락을 반영한 현실적 모델 성능
- AdaBoost: 데이터가 불균형할 때 오분류에 민감한 특성 (Boosting 특유의 과적합 영향). 다수 클래스(Foot)에 집중하고 소수 클래스(Other, Head)는 상대적으로 놓침



03. 숫 위치에 따른 숫 타입 분류 모델

최종 own구조

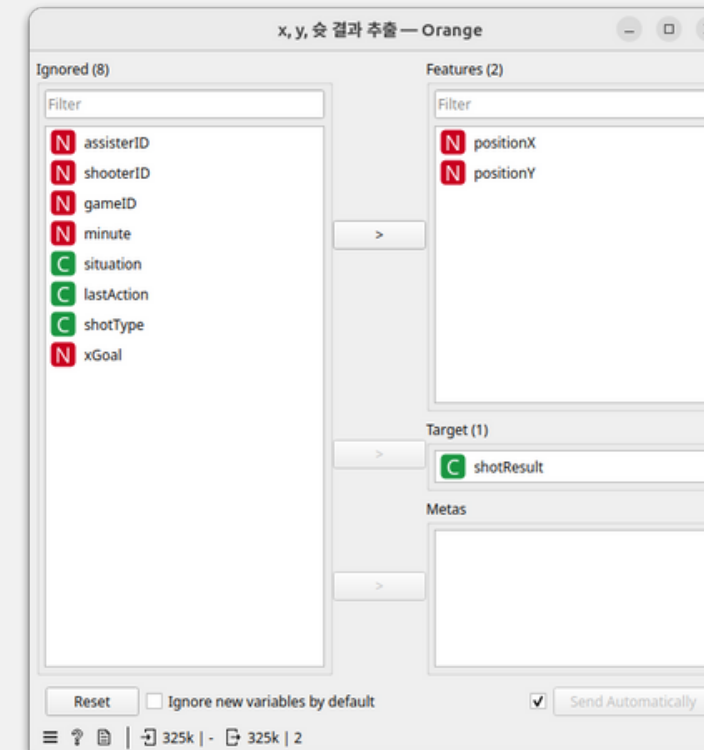


04. 슈트 위치(positionX, Y)에 따른 골 분류 모델

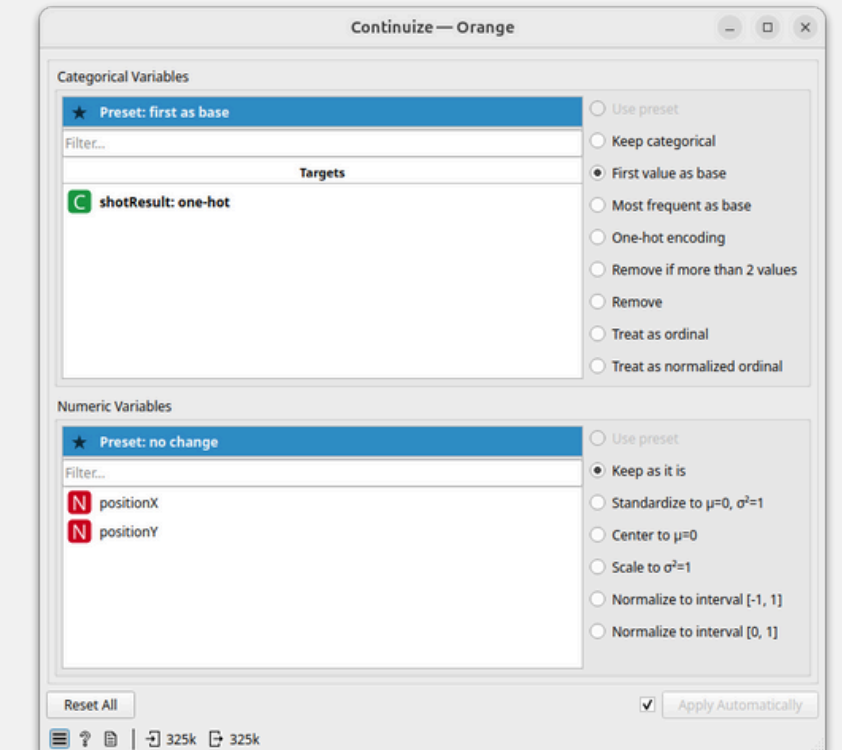
데이터 전처리

- 이상치 없음, 결측치 있지만 모델에 불필요한 열에 존재
→ 별도 처리 불필요
 - posX, posY, shotResult 제외 전부 제거
- shotResult 단순화
 - 기존값: 선방(BlockedShot), 골(Goal), 자책골(Own Goal) 등 여러 카테고리
 - One-hot encoding 사용하여 숫자형으로 변환
 - 골과 노골 외의 데이터는 모두 제거 후 숫자형 데이터 범주화
- 최종 형태
 - 입력 변수: PositionX, PositionY
 - 목표 변수: ShotResult (범주형 0 or 1)

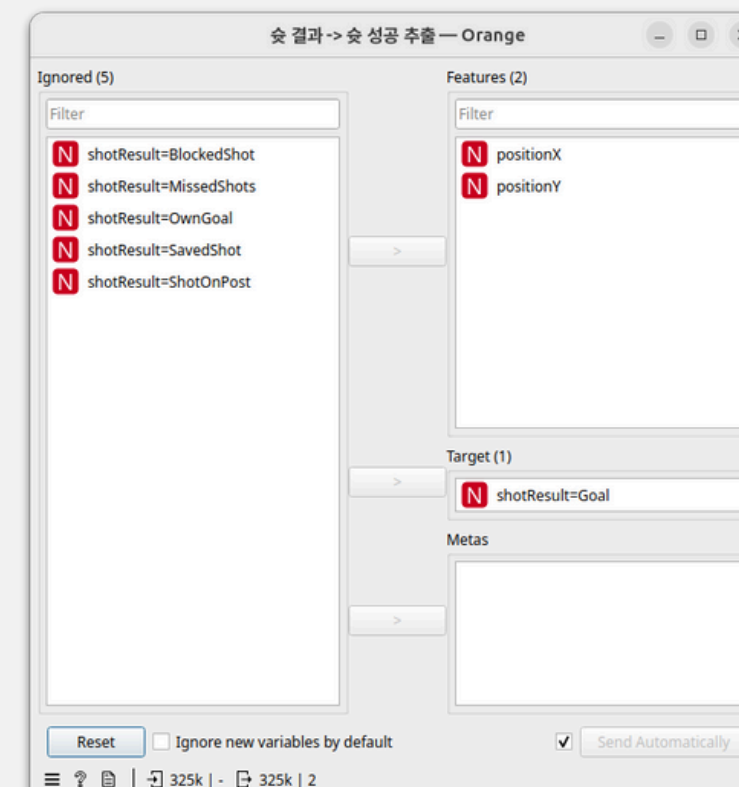
※ Orange3에서 Target이 숫자형일 경우 회귀로 인식, 분류를 위해 범주형으로 변환 필요



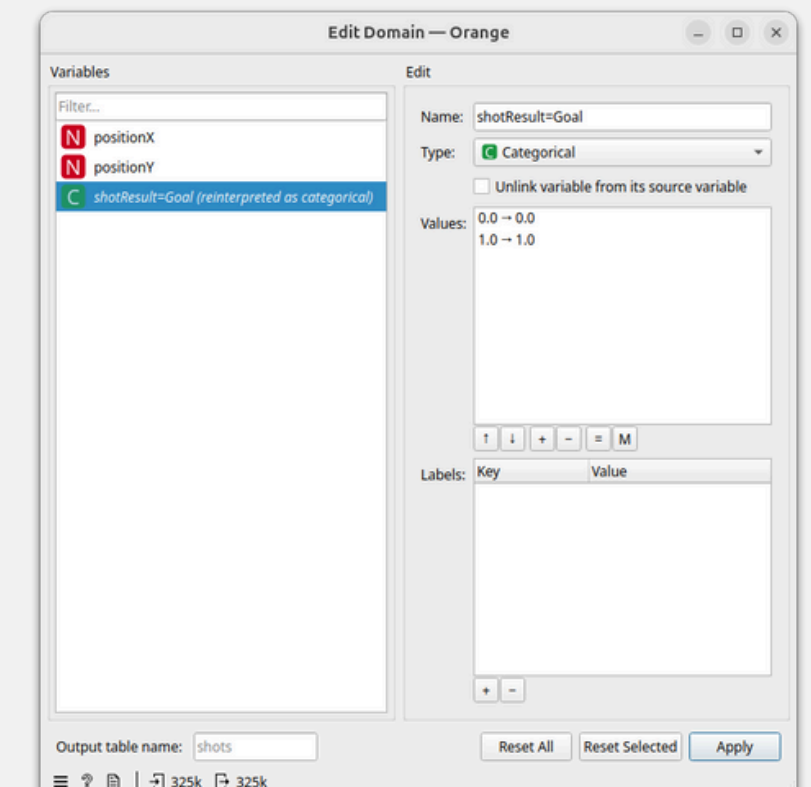
1. 타겟 Columns 설정



2. 범주형 변수 인코딩



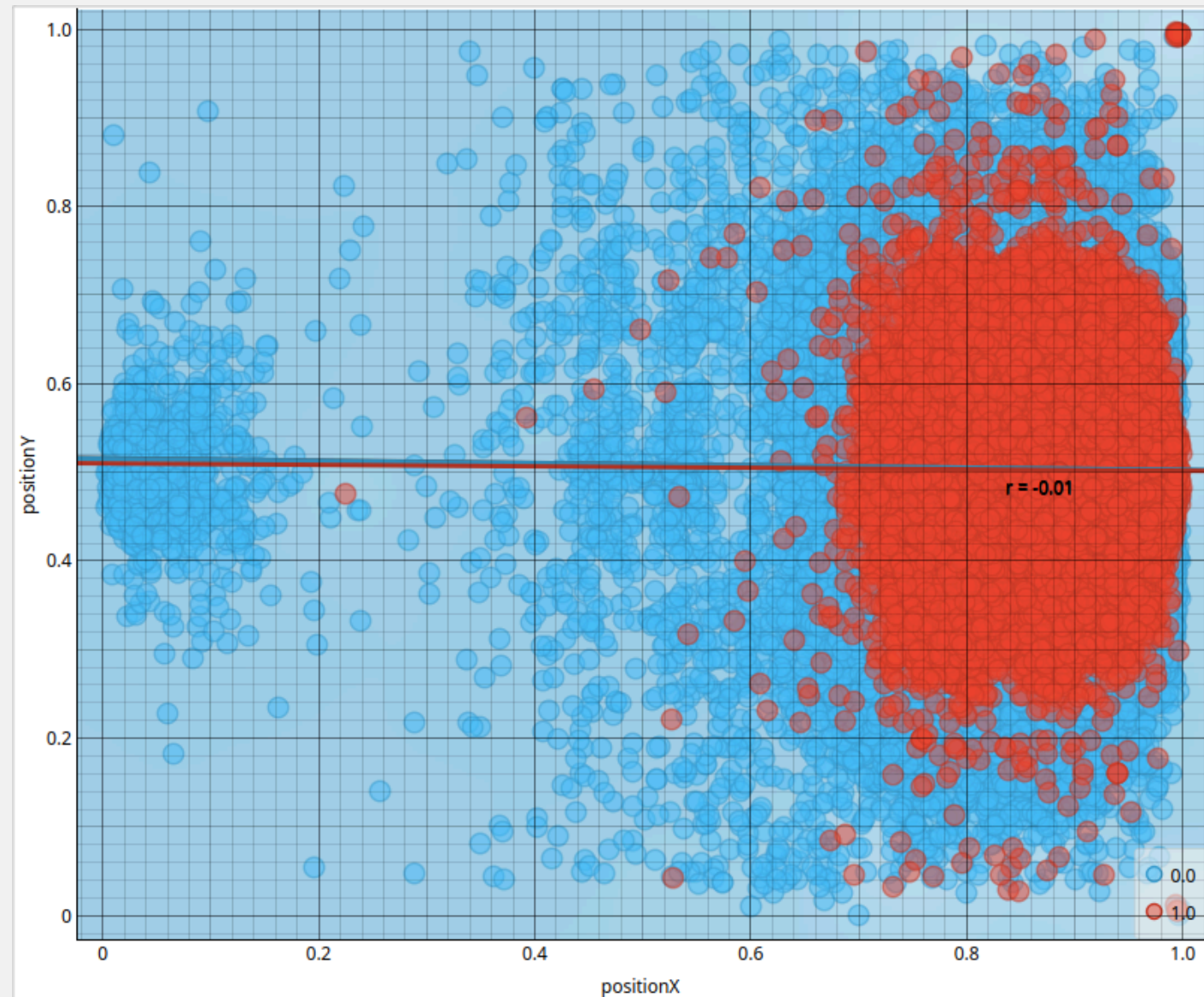
3. 슈트 결과->슈트 성공 추출



4. 숫자형 데이터 범주형 변환

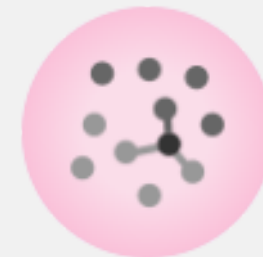
04. 슛 위치(positionX, Y)에 따른 골 분류 모델

슛 위치에 따른 골 성공 Scatter plot



머신러닝 모델 구축

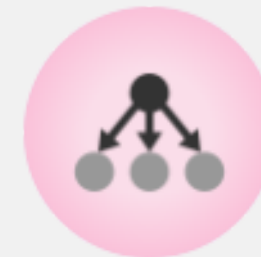
- kNN - 수치 기반 데이터 분류 작업에서 성능이 우수
- Neural Network - 비선형 관계나 복잡한 패턴을 학습 가능
- Naive Bayes - 사전확률정보 기반 사후확률 추정 가능
- Tree - 규칙 기반 분류
- Stack - 여러 모델의 장점을 결합해 예측 성능 향상



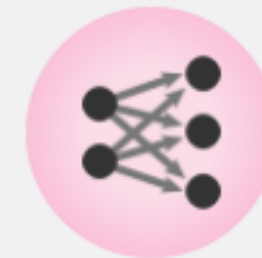
kNN



Tree



Naive Bayes



Neural Network



Stacking

04. 슈트 위치(positionX, Y)에 따른 골 분류 모델

결과

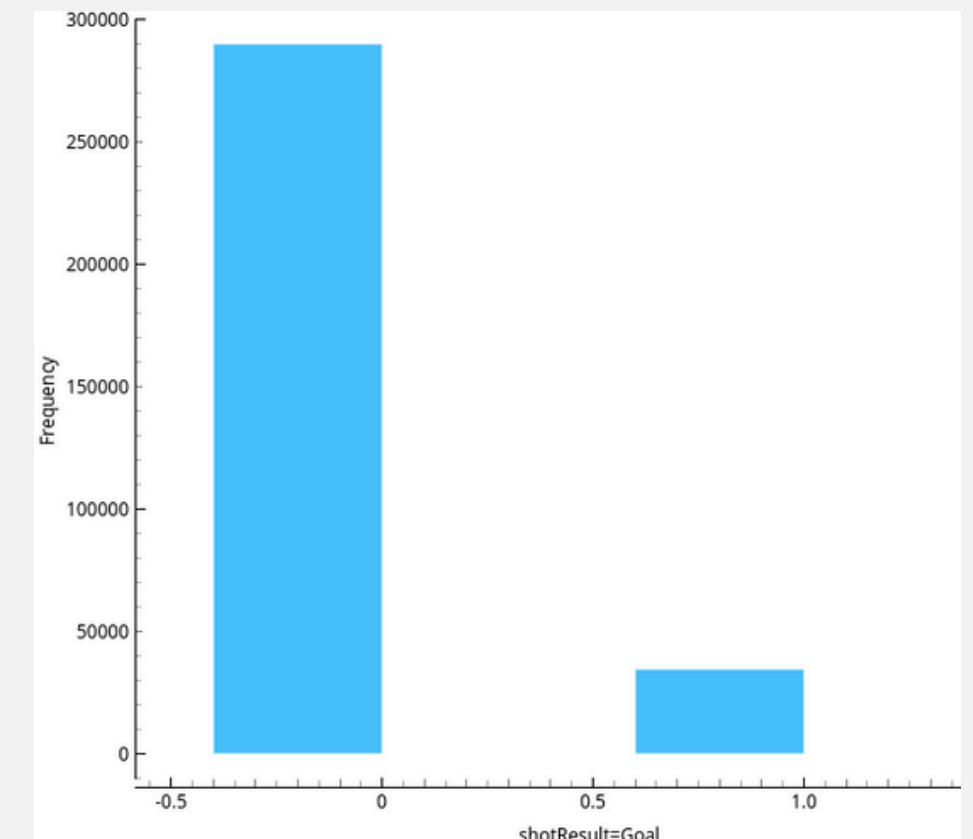
- 분류 모델들의 성능 종합을 봤을 때, 예측은 꽤나 준수
- 데이터의 심한 불균형(약 1:9)으로 인해, 모델이 노골 클래스로 편향되어 실제 예측력보다 높은 정확도를 보임
- Confusion matrix를 통해서 모델에 문제가 있다는 것을 알 수 있음

정확도 저하 요인 분석

- 골 데이터와 노골 데이터 불균형
 - 데이터 리샘플링을 통해 데이터 균형 맞추기
 - 데이터 확장을 통해 다른 경기, 시즌, 리그 데이터를 추가
- 기본 임계값(0.5)이 데이터의 실제 분포와 맞지 않아, 골 예측 시 지나치게 보수적으로 작동함.

Model	AUC	CA	F1	Prec	Recall	MCC
kNN	0.763	0.903	0.875	0.887	0.903	0.303
Tree	0.762	0.903	0.874	0.887	0.903	0.299
Neural Network	0.768	0.902	0.872	0.882	0.902	0.280
Naive Bayes	0.734	0.894	0.844	0.799	0.894	0.000
Stack	0.770	0.903	0.878	0.884	0.903	0.317

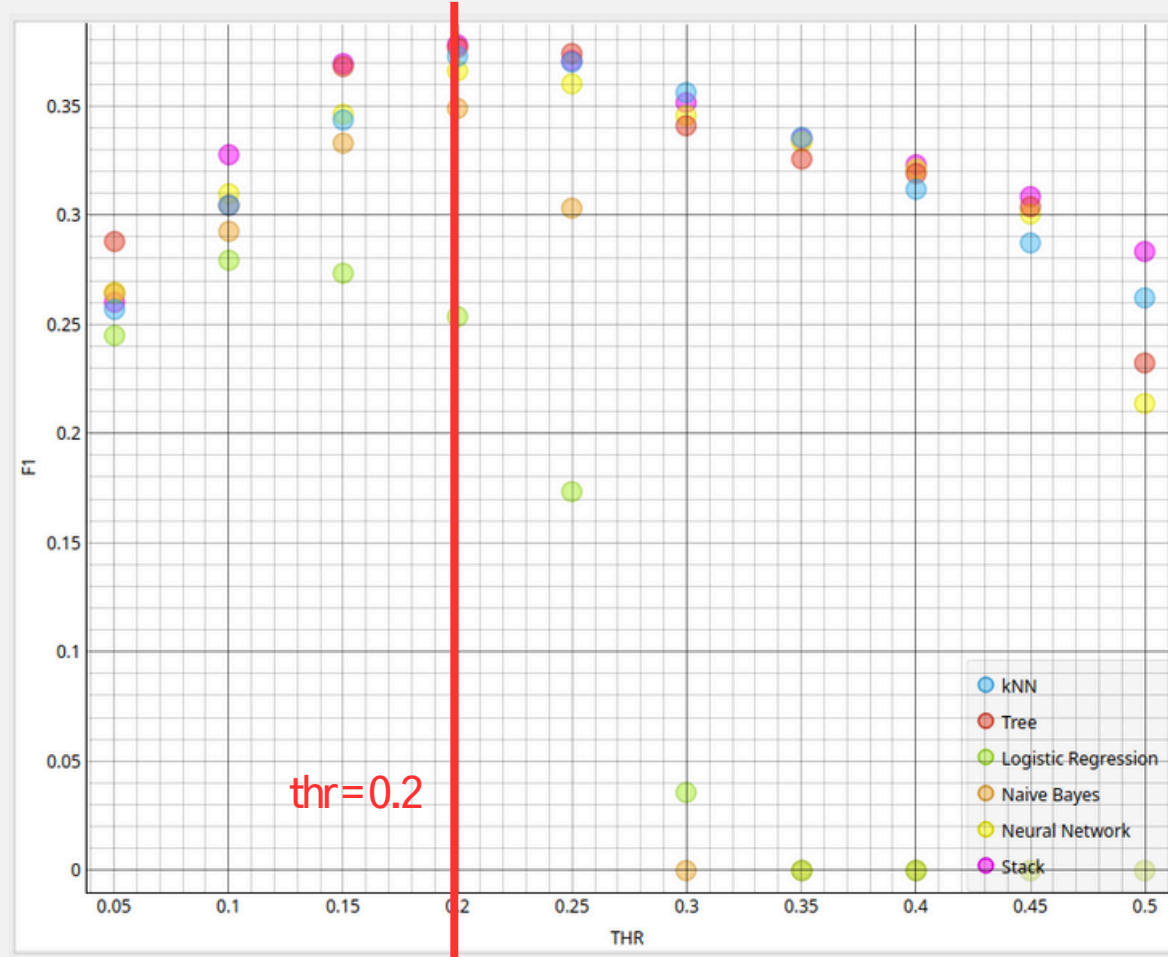
		Predicted		Σ
		0.0	1.0	
Actual	0.0	861166	8974	870140
	1.0	85052	18438	103490
Σ		946218	27412	973630



04. 슛 위치(positionX, Y)에 따른 골 분류 모델

개선사항

- 모델 사용을 위해 골 탐지 성능 향상 필요
- 슛에 대한 골과 노골의 비율은 약 1 : 8.9
- 임계값 (Threshold=0.5)를 조절함으로써 과적합 해소, 성능 최적화



- Threshold = 0.2 ~ 0.22일 때 최대 F1 score 출력

개선

- 모델 별 최적 Threshold 값 (F1 score 기준)

	Model	Best_THR	Max_F1	Accuracy	Precision	Recall	TP	FP	TN	FN
6	Stack	0.2	0.377808	0.872931	0.39393	0.362953	37562	57790	812350	65928
2	Tree	0.2	0.376492	0.875334	0.401904	0.354102	36646	54535	815605	66844
1	kNN	0.22	0.374199	0.876077	0.403903	0.348565	36073	53238	816902	67417
5	Neural Network	0.22	0.366343	0.871905	0.386281	0.348362	36052	57279	812861	67438
4	Naive Bayes	0.2	0.348551	0.786141	0.257722	0.538245	55703	160433	709707	47787
3	Logistic Regre...	0.12	0.292877	0.640553	0.185155	0.700309	72475	318953	551187	31015

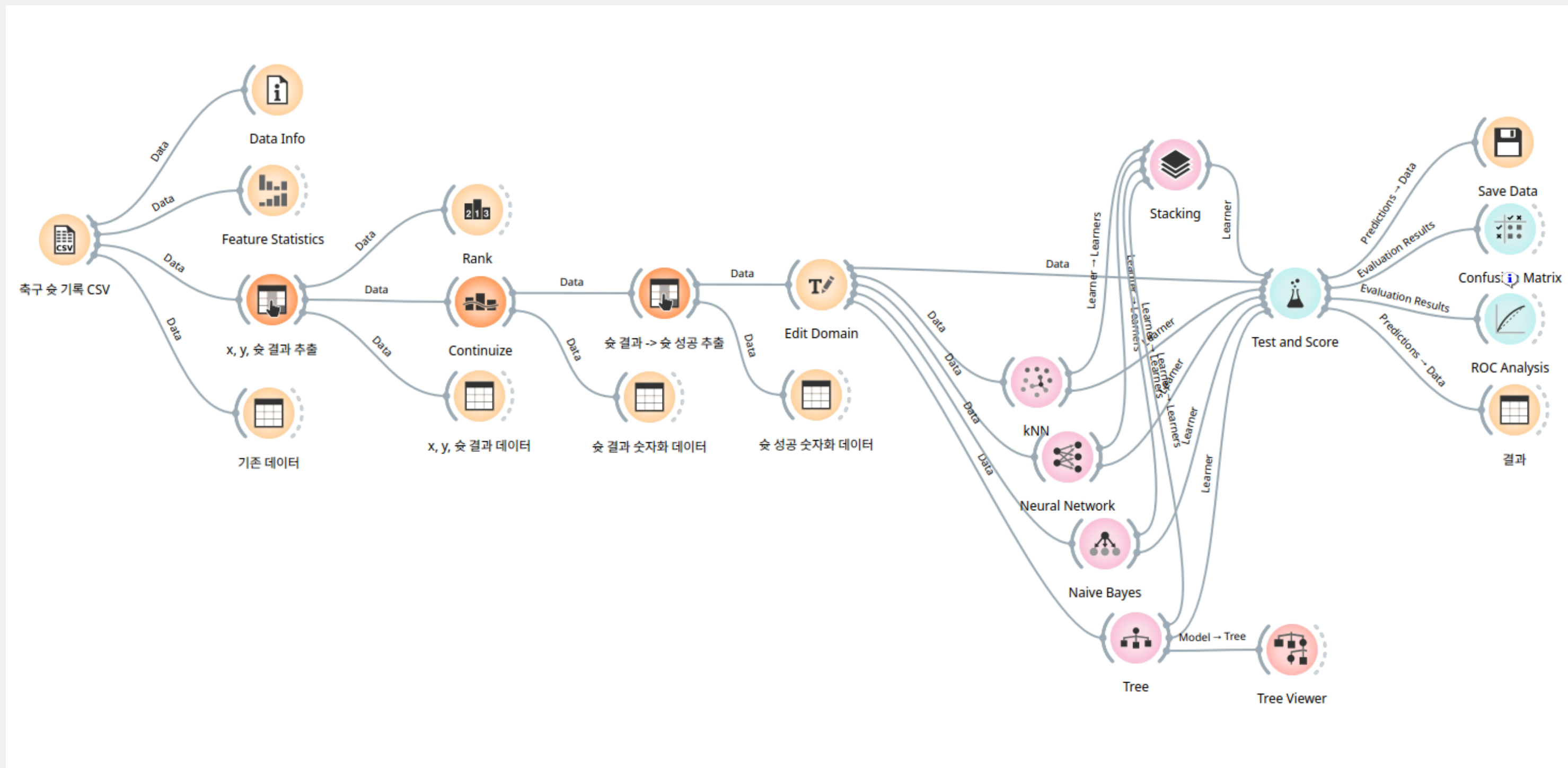
- 기존 Threshold = 5와 Threshold = 2와 비교 (Stack 모델)

Metric	Thr = 0.5 (기존)	Thr = 0.2 (개선)	변화	설명
TP	18,438	37,562	103.72%	실제 골 판정 증가
FP	8,974	57,790	543.97%	가짜 골 판정 증가
TN	861,166	812,350	-5.67%	실제 노골 판정 감소
FN	85,052	65,928	-22.49%	가짜 노골 판정 감소
Accuracy	0.902	0.873	-3.22%	정확도 감소
Precision	0.673	0.393	-41.60%	정밀도 하락
Recall	0.178	0.363	103.93%	탐지율 증가
F1 score	0.274	0.378	37.96%	균형 개선

- 정밀도는 떨어졌지만 모델의 골탐지성능 향상, F1 Score 개선

04. 슛 위치(positionX, Y)에 따른 골 분류 모델

최종 own구조



05. 결론

슛 위치에 따른 슛 타입 분류 모델

- positionX, positionY(슛 위치 좌표)는 슛 타입(Head/Foot/Other)을 구분하는 가장 핵심적인 공간적 변수로 작용
- Head 슛은 골문 근처·공중볼 상황(Aerial, Cross) 에서 주로 발생했고, Foot 슛은 오픈플레이 중 중거리 지역에서 빈번히 발생
- 따라서, 슛의 공간적 패턴(spatial pattern) 과 행동 맥락(lastAction) 의 결합이 슛 타입을 결정짓는 주요 요인임을 확인

슛 위치에 따른 골 분류 모델

- 위치 정보만으로도 일정 수준의 예측력을 보였지만, 실제 데이터에서 골보다 노골이 약9배 많아 모델이 한쪽으로 편향되는 문제가 있었음
- 데이터 임계값(Threshold)를 0.5에서 0.2로 수정, 골 탐지 성능 및 F1 score는 향상 되나 정밀도가 떨어짐

향후 방향

- 슛 좌표+각도+슈팅 속도 등 공간·물리적 특성 확장 모델링
 - 슛 위치에 따른 골 분류 모델 확률을 보정 및 정교화
- 시각화 기반 Shot Map 시스템 구축 → 위치별 슛 분포 및 성공 확률 시각 분석
- 향후에는 선수별 슛 성향을 반영한 개인화된 슈팅 성과 모델로 확장 가능



감사합니다

SeSAC
인텔 Smart AI master 2기

피톤치드

박현욱 | 성시경 | 염수림
