

## 학습데이터

Height (m)	Favorite Color	Gender	Weight (kg)
1.6	Blue	Male	88
1.6	Green	Female	76
1.5	Blue	Female	56
1.8	Red	Male	73
1.5	Green	Male	77
1.4	Blue	Female	57

- 키, 좋아하는 색, 성별로 몸무게를 예측하는 모델을 생성
- feature: Height(m), Favorite Color, Gender
- target: Weight(kg)

Average Weight

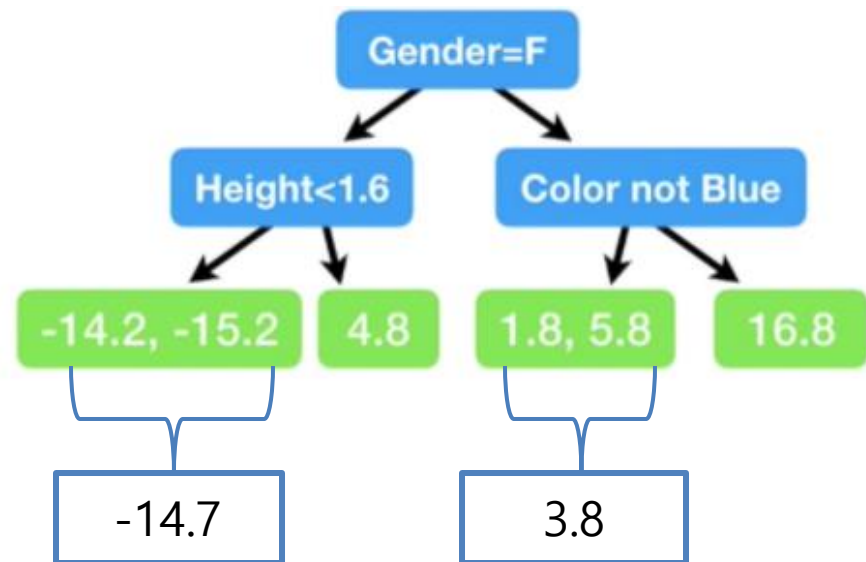
71.2

Height (m)	Favorite Color	Gender	Weight (kg)	Residual
1.6	Blue	Male	88	16.8
1.6	Green	Female	76	4.8
1.5	Blue	Female	56	-15.2
1.8	Red	Male	73	1.8
1.5	Green	Male	77	5.8
1.4	Blue	Female	57	-14.2

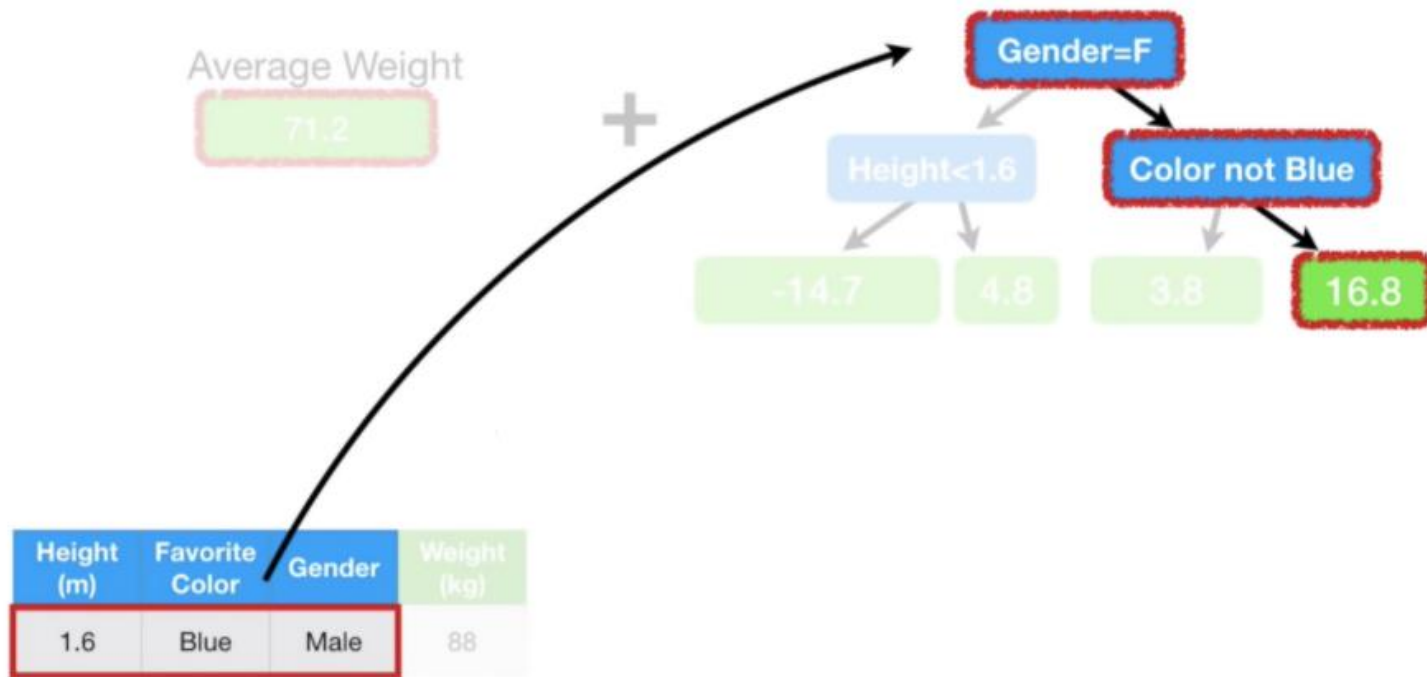
- 첫 모델은 Target을 예측한다.
- Residual(잔차) = 실제 Weight - 예측한 Weight
  - 잔차는 오차와 같은 의미

Height (m)	Favorite Color	Gender	Weight (kg)	Residual
1.6	Blue	Male	88	16.8
1.6	Green	Female	76	4.8
1.5	Blue	Female	56	-15.2
1.8	Red	Male	73	1.8
1.5	Green	Male	77	5.8
1.4	Blue	Female	57	-14.2

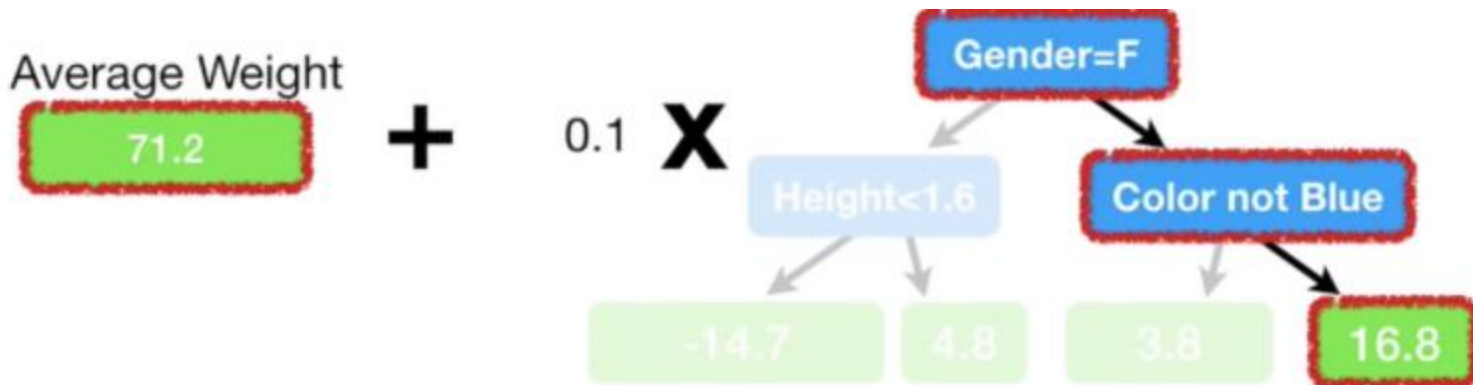
Residual을 예측하는  
트리모델 생성



- 두 번째 트리모델부터는 잔차를 예측하는 모델을 Feature로 잔차(Residual)를 예측하는 트리모델을 생성한다.



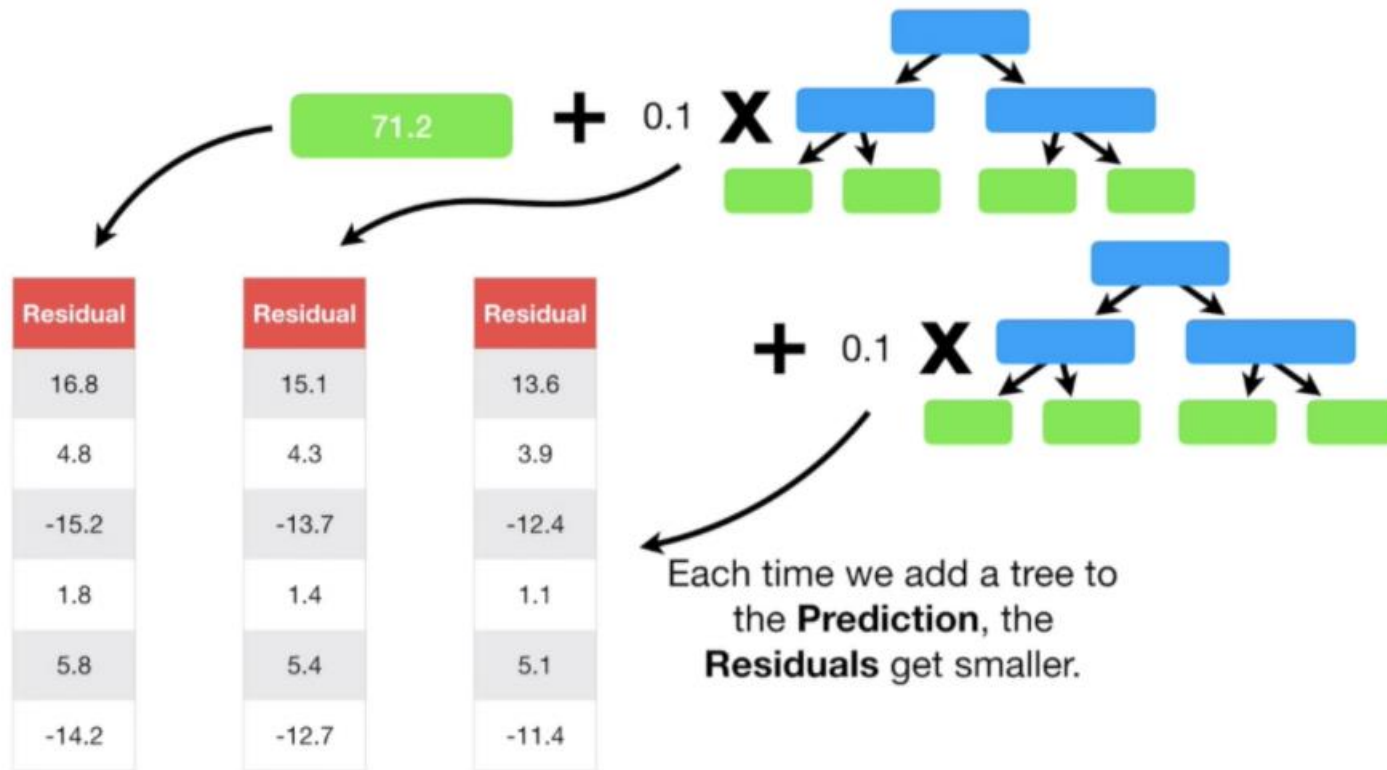
- 관측치를 트리 모델로 잔차(오차)를 예측 한다.
- 위 그림에서 첫 번째 관측치의 경우 잔차가 16.8이 나온다. 그럼 그것을 첫 번째 예측 값이었던 평균인 71.2에 더하면 실제 값인 88이 나온다.



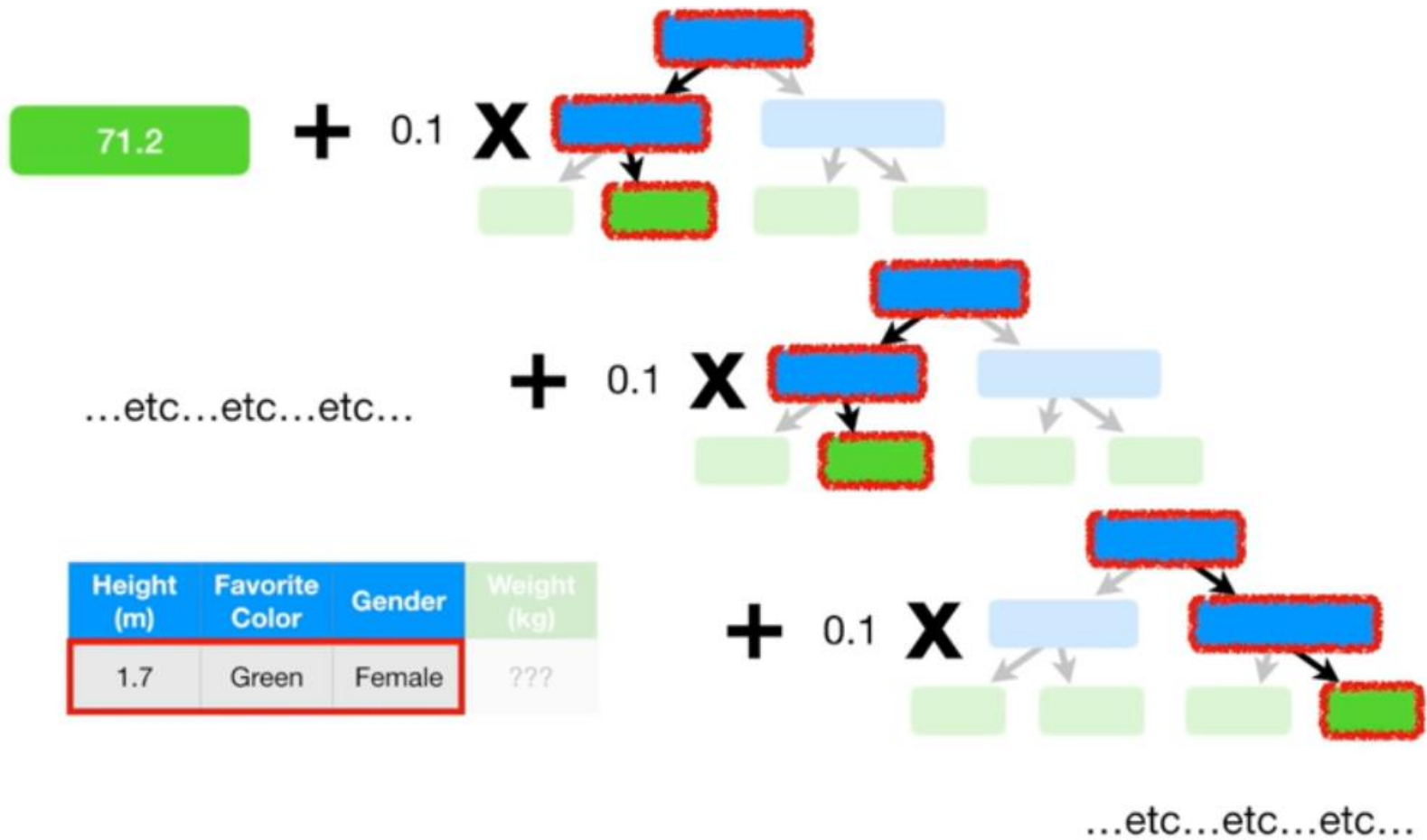
Now the **Predicted Weight** =  $71.2 + (0.1 \times 16.8) = 72.9$

Height (m)	Favorite Color	Gender	Weight (kg)
1.6	Blue	Male	88

- 예측한 잔차를 그대로 더하면 학습데이터의 값은 100% 예측하겠지만 새로운 데이터에는 맞지 않을 가능성이 높다. (과대적합)
- 그래서 예측한 잔차(오차)에 학습율을 곱한 값을 예측 값에 더한다.
- 학습율(Learning Rate) : 하이퍼 파라미터로 오차가 있는 예측값을 보정하는 비율.
- 위의 예를 보면 처음 예측한 71.2가 72.9로 실제 값에 좀더 가까이 예측했다.



- 줄어든 잔차를 예측하는 트리모델을 만들어 예측한 결과에 학습율을 곱 한 뒤 이전 예측 결과에 더한다.
- 예측값 + 학습율\*잔차예측값 = 새데이터
- 위의 작업을 반복하여 잔차를 줄여 나간다.



- 새로운 데이터 예측
  - 생성된 트리모델들을 거치면서 결과를 예측한다.

- 이미지 참조
- [https://www.youtube.com/watch?v=3CC4N4z3GJc&list=PLblh5JKOoLUICTaGLRoHQDuF\\_7q2GfuJF&index=49](https://www.youtube.com/watch?v=3CC4N4z3GJc&list=PLblh5JKOoLUICTaGLRoHQDuF_7q2GfuJF&index=49)