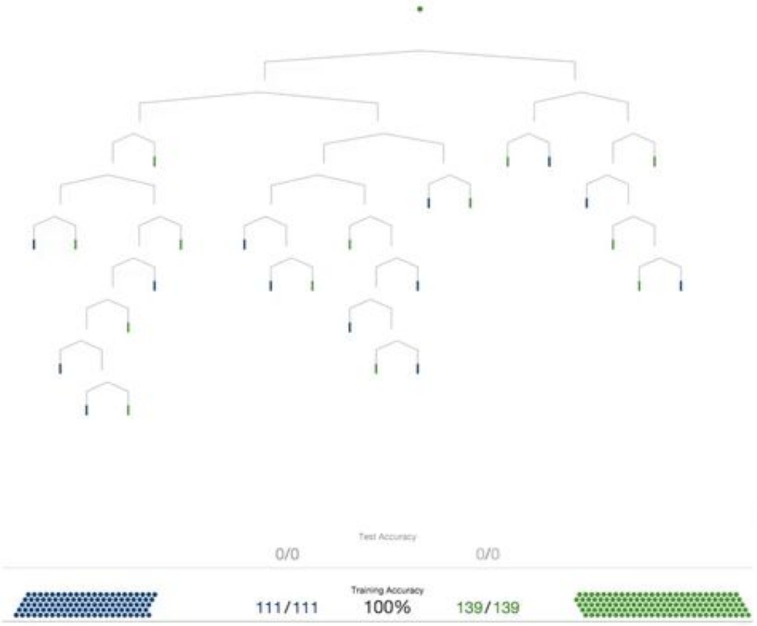
How feature importances are calculated in a Random Forest classifier

Benjamin Chew
Presentation for Growth Intelligence



Decision Tree (Toy Example)

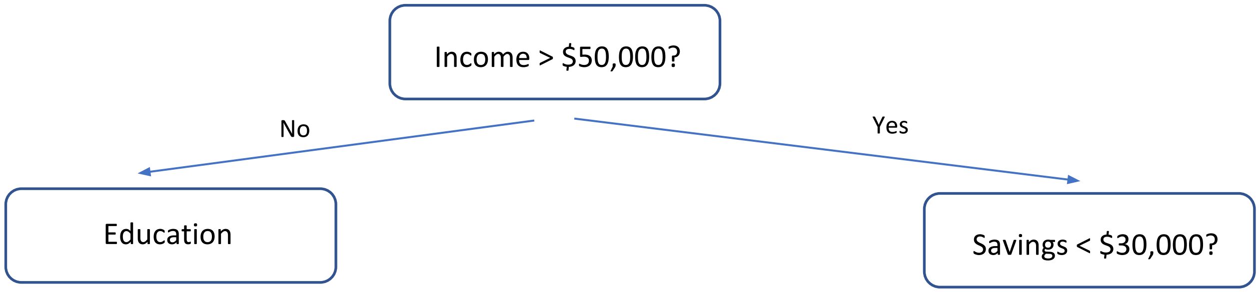
Imagine that you are working for a bank and are deciding whether to give people a loan (APPROVED/DENIED) based on the following features: *Income*, *Education*, and *Savings*.

Toy Example Credit: David Maillet

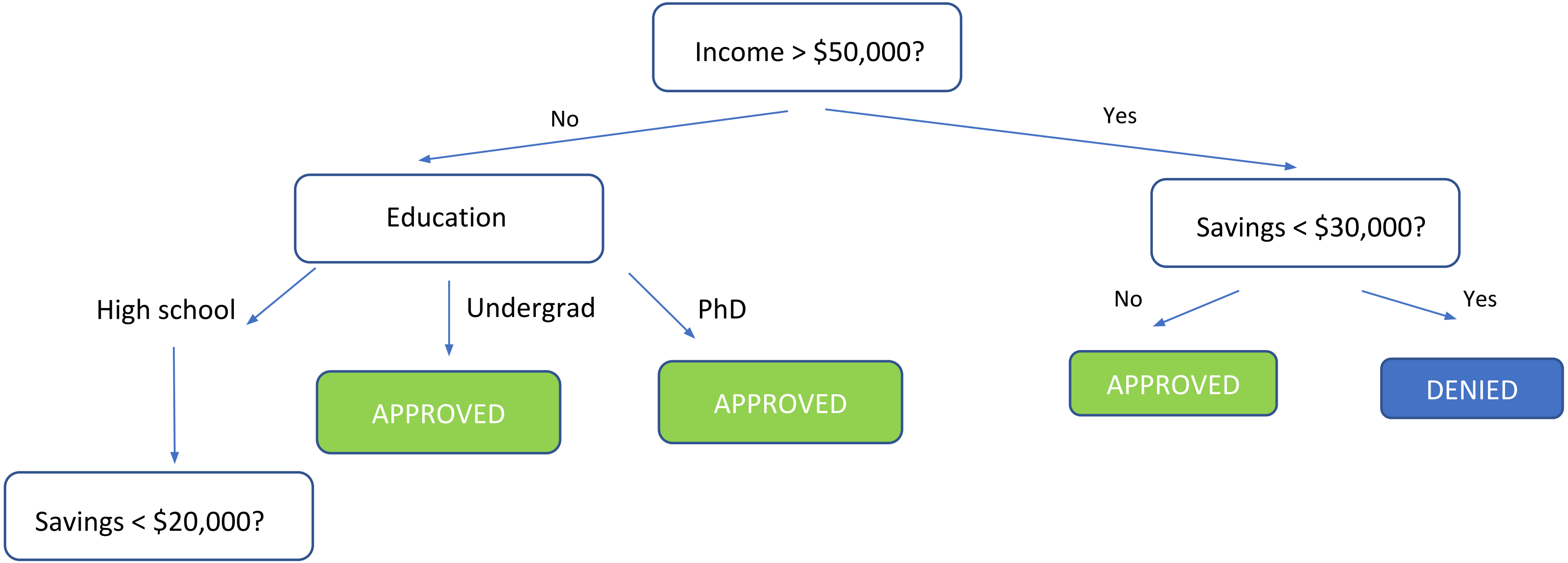
Decision Tree (Toy Example)

Income > $50,000?

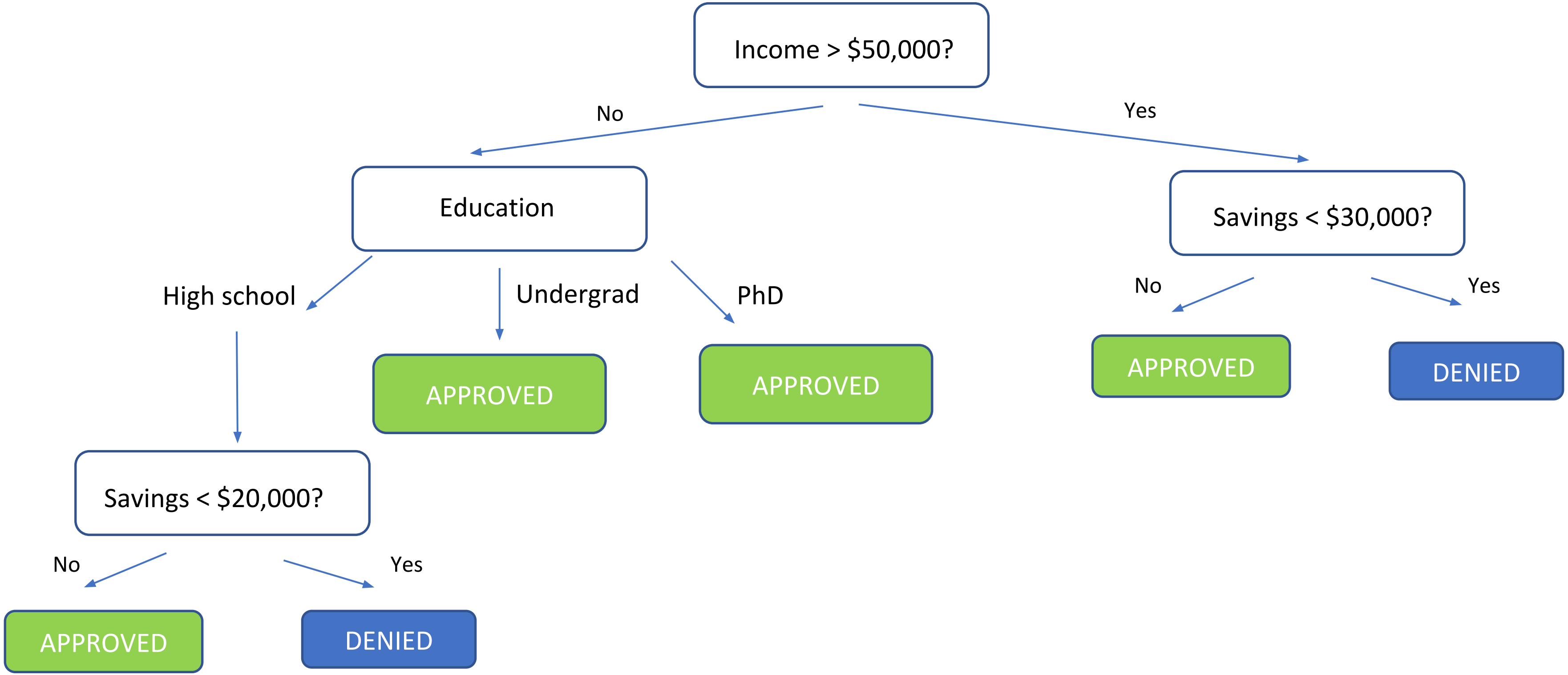
Decision Tree (Toy Example)



Decision Tree (Toy Example)



Decision Tree (Toy Example)



Random Forests

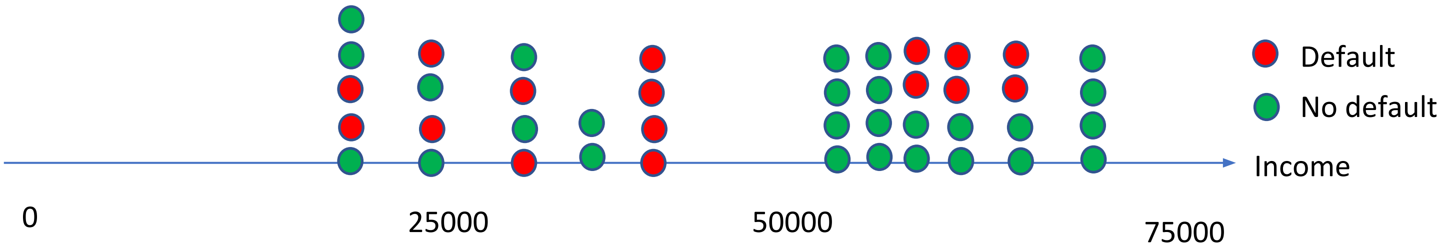
- Decision trees built on bootstrapped training samples, *N*
- Random sample of predictors chosen as split candidates
- Stop when minimum criteria reached (e.g. no further reduction in impurity)

$$I_G(n) = 1 - \sum_{i=1}^{J} (p_i)^2$$

Mean Decrease Impurity (MDI)

- Default method used in scikit-learn
- Maximize the decrease of some impurity measure

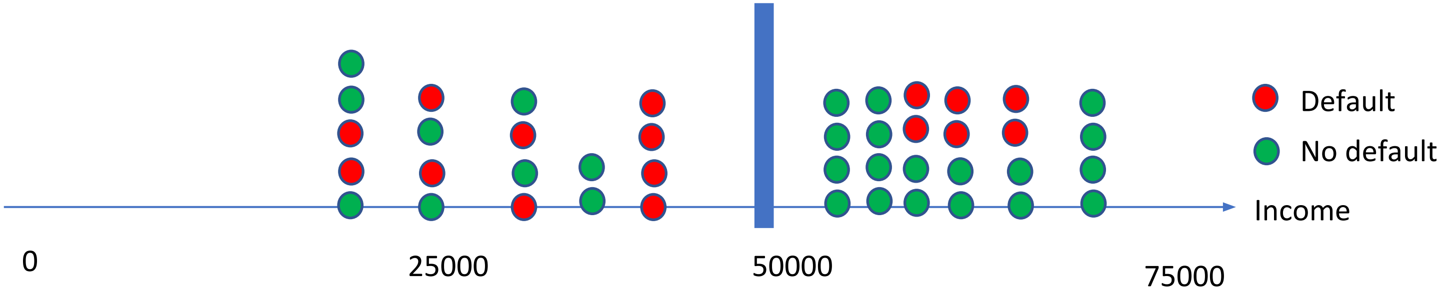
$$\Delta i(s,t) = i(t) - p_L i(t_L) - p_R i(t_R) \quad \textit{where} \quad p_L = N_{t_L}/N_t \quad \textit{and} \quad p_R = N_{t_R}/N_t$$



Mean Decrease Impurity (MDI)

- Default method used in scikit-learn
- Maximize the decrease of some impurity measure

$$\Delta i(s,t) = i(t) - p_L i(t_L) - p_R i(t_R) \quad \textit{where} \quad p_L = N_{t_L}/N_t \quad \textit{and} \quad p_R = N_{t_R}/N_t$$



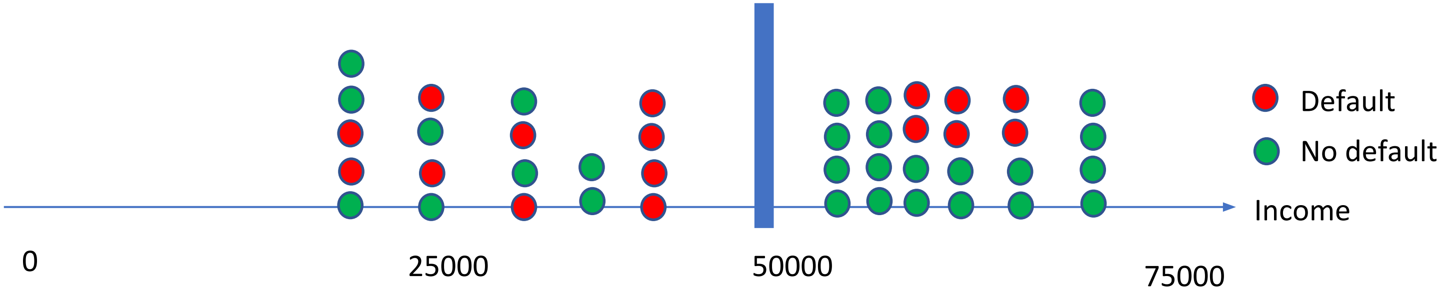
Mean Decrease Impurity (MDI)

- Default method used in scikit-learn
- Maximize the decrease of some impurity measure

$$\Delta i(s,t) = i(t) - p_L i(t_L) - p_R i(t_R) \quad \textit{where} \quad p_L = N_{t_L}/N_t \quad \textit{and} \quad p_R = N_{t_R}/N_t$$

$$= 0.463 - (20/44)*0.5 - (24/44)*0.375$$

$$= 0.031$$



Mean Decrease Impurity (MDI)

- Default method used in scikit-learn
- Maximize the decrease of some impurity measure

$$\Delta i(s,t) = i(t) - p_L i(t_L) - p_R i(t_R) \quad \textit{where} \quad p_L = N_{t_L}/N_t \quad \textit{and} \quad p_R = N_{t_R}/N_t$$

$$\mathrm{Imp}(X_j) = \frac{1}{M} \sum_{m=1}^{M} \sum_{t \in \varphi_m} 1(j_t = j) \left[p(t) \Delta i(s_t, t) \right]$$

Mean Decrease Impurity (MDI)

- Default method used in scikit-learn
- Maximize the decrease of some impurity measure

$$\Delta i(s,t) = i(t) - p_L i(t_L) - p_R i(t_R) \quad \textit{where} \quad p_L = N_{t_L}/N_t \quad \textit{and} \quad p_R = N_{t_R}/N_t$$

$$\mathrm{Imp}(X_j) = \frac{1}{M} \sum_{m=1}^{M} \sum_{t \in \varphi_m} 1(j_t = j)\Big[p(t)\Delta i(s_t, t)\Big]$$

- Total decrease in node impurity, weighted by the probability of reaching that node, averaged over all trees of the ensemble

Mean Decrease Accuracy (MDA) or *Permutation Importance*

- Out-of-Bag (OOB) samples

Employed	Marital Status	Education	Loan Default
Yes	Married	High School	No
No	Single	Bachelors	Yes
Yes	Single	Masters	Yes
Yes	Single	Bachelors	No
No	Married	Bachelors	No

Mean Decrease Accuracy (MDA) or *Permutation Importance*

- Out-of-Bag (OOB) samples

Employed	**Marital Status**	**Education**	**Loan Default**
Yes	Married	High School	No
No	Single	Bachelors	Yes
Yes	Single	Masters	Yes
Yes	Single	Bachelors	No
No	Married	Bachelors	No

Bootstrapped Samples Tree A

Employed	**Marital Status**	**Education**	**Loan Default**
Yes	Married	High School	No
No	Single	Bachelors	Yes
Yes	Married	High School	No
No	Married	Bachelors	No
Yes	Married	High School	No

Mean Decrease Accuracy (MDA) or *Permutation Importance*

- Out-of-Bag (OOB) samples

Employed	**Marital Status**	**Education**	**Loan Default**
Yes	Married	High School	No
No	Single	Bachelors	Yes
Yes	Single	Masters	Yes
Yes	Single	Bachelors	No
No	Married	Bachelors	No

Bootstrapped Samples Tree A

Employed	**Marital Status**	**Education**	**Loan Default**
Yes	Married	High School	No
No	Single	Bachelors	Yes
Yes	Married	High School	No
No	Married	Bachelors	No
Yes	Married	High School	No

Mean Decrease Accuracy (MDA) or *Permutation Importance*

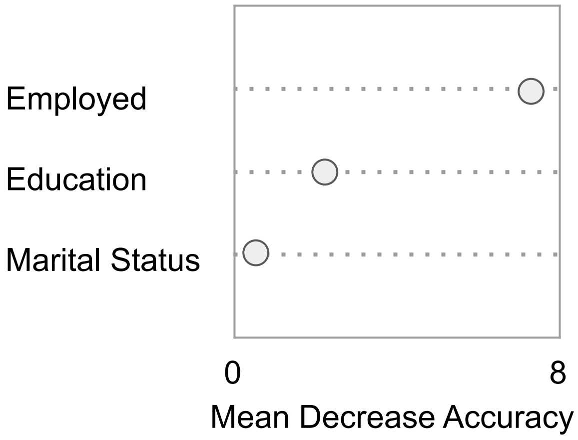
- Out-of-Bag (OOB) samples
- Calculate OOB score: number of correctly predicted rows from OOB sample

Employed	**Marital Status**	**Education**	**Loan Default**
Yes	Single	Bachelors	No

Tree	**Prediction**
A	YES
C	NO
D	NO
Majority Vote: NO	

Mean Decrease Accuracy (MDA) or *Permutation Importance*

- Out-of-Bag (OOB) samples
- Calculate OOB score: number of correctly predicted rows from OOB sample
- Randomly permute the values for a single predictor and record the change in performance



Drop-column Importance

1. Calculate baseline performance
2. Remove a column (predictor)
3. Train the reduced model
4. Recompute performance score

$$Imp(X_j) = Score_{\text{Baseline Model}} - Score_{\text{Reduced Model}}$$

Things to Note

- Mean Decrease Impurity
  - "*the variable importance measures of Breiman's original Random Forest method ... are not reliable in situations where potential predictor variables vary in their **scale of measurement** or their **number of categories***." (Strobl et al, 2007)

- Mean Decrease Accuracy
  - "*permutation importance over-estimates the importance of correlated predictor variables.*" (Strobl et al, 2008)
  - Conditional permutation scheme

- Drop-column Importance
  - Computationally expensive

Conditional Permutation Importance

$H_0 : X_j \perp Y, Z$

Y	X_j	Z
y_1	$x_{\pi_j(1),j}$	z_1
$\vdots$	$\vdots$	$\vdots$
y_i	$x_{\pi_j(i),j}$	z_i
$\vdots$	$\vdots$	$\vdots$
y_n	$x_{\pi_j(n),j}$	z_n

$$P(Y, X_j, Z) \underset{H_0}{=} P(Y, Z) \cdot P(X_j).$$

$H_0 : (X_j \perp Y) | Z$

Y	X_j	Z
y_1	$x_{\pi_{j\|Z=a}(1),j}$	$z_1 = a$
y_3	$x_{\pi_{j\|Z=a}(3),j}$	$z_3 = a$
y_{27}	$x_{\pi_{j\|Z=a}(27),j}$	$z_{27} = a$
y_6	$x_{\pi_{j\|Z=b}(6),j}$	$z_6 = b$
y_{14}	$x_{\pi_{j\|Z=b}(14),j}$	$z_{14} = b$
y_{21}	$x_{\pi_{j\|Z=b}(21),j}$	$z_{21} = b$
$\vdots$	$\vdots$	$\vdots$

$$P(Y, X_j \mid Z) \overset{H_0}{=} P(Y \mid Z) \cdot P(X_j \mid Z)$$

or $$P(Y \mid X_j, Z) \overset{H_0}{=} P(Y \mid Z),$$

1. Compute OOB-prediction accuracy for each tree
2. Determine Z, the number of variables to be conditioned on
3. For all Z, take the points that split the variable in the current tree
4. Create a grid by bisecting the sample space at each split point
5. Within each grid, permute the values of X_j and compute the OOB-prediction accuracy
6. Imp(X_j) is the difference between prediction accuracy before and after permutation, averaged over all trees