The cox PH model has the form:

$$\frac{h(t)}{h_0(t)} = \prod_{i=1}^{m} \exp\left(\beta^T x_i\right)$$

β^T : importance score

x_i : feature vector for instance i

cox regression models will solve for the β coefficients by maximizing the partial likelihood function

$$L(\beta) = \prod_{i,\delta_i=1} \frac{\exp\{\beta^T x_i\}}{\sum_{j \in R(t_i)} \exp\{\beta^T x_j\}}$$

where the product is over all event times

$\Rightarrow$ the sum includes all individuals in the risk set $R(t_i)$ (individuals for which the event has <u>not</u> occurred)

Note: the numerator does <u>not</u> include censored individuals while the denominator does

$L(\beta)$ can be linearized by taking the log of both sides

$$\Rightarrow \ell(\beta) = \sum_{i,\delta_i=1} \left[\underbrace{\beta^T x_i}_{\text{"hits"}} - \underbrace{\ln\left(\sum_{j \in R(t_j)} \exp\left(\beta^T x_j\right)\right)}_{\text{"misses" or normalization factor}} \right]$$

Relief handles approximating feature importance differently...

1. Find the neighborhood

$$\begin{bmatrix} R_1\ NN_1 \\ R_1\ NN_2 \\ \vdots \\ R_n\ NN_1 \\ R_n\ NN_2 \\ R_n\ NN_3 \end{bmatrix}$$

2. Calculate distance between R_i & each of its neighbor hits $\Delta_{\#}$ & each of its neighbor misses Δ_m

3. we end up with the updr matrix with rows representing each R_i, NN pair

NN_1	Δ_{att}	Δ_{phen}
NN_2	Δ_{att}	Δ_{phen}
NN_3		

4. If the output is binomial (1: hit, 0: miss)

then we perform binomial regression

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n$$

∴ the loss function is the following:

$$L(\beta) = \prod_{i=1}^{m} p(X_i)^{Y_i} \left(1 - p(X_i)\right)^{(1-Y_i)}$$

where in our case

X_i : Δ_{att_i}

Y_i : Δ_{phen_i}

$$\ell(\beta) = \sum_{i=1}^{m} \left[Y_i \log\left(p(X_i)\right) + (1 - Y_i) \log\left(1 - p(X_i)\right) \right]$$

where $p(X_i) = \dfrac{1}{1 + \exp\left(-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots)\right)}$

So to sum it up ... cox regression trees to maximize

$$\ell(\beta) = \sum_{i,\delta_i=1} \left[\underbrace{\beta^T x_i}_{\text{"hits"}} - \underbrace{\ln\left(\sum_{j \in R(t_j)} \exp\left(\beta^T x_j\right) \right)}_{\text{"misses" or normalization factor}} \right]$$

where the summation takes place over the entire "at risk" set
& "hits" are only individuals for which the event has happened

while updr trees to minimize

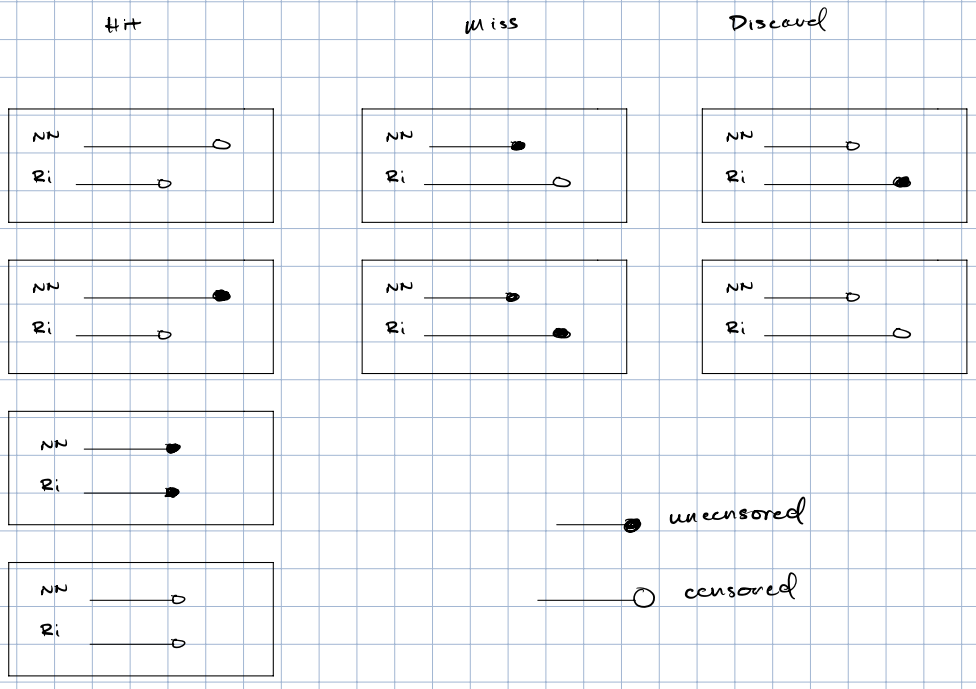
$$\ell(\beta) = \sum_{i=1}^{m} \left[Y_i \log\left(p(x_i)\right) + (1 - Y_i) \log\left(1 - p(x_i)\right) \right]$$

$$\text{where } p(x_i) = \frac{1}{1 + \exp\left(-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots)\right)}$$

where the summation takes place over the neighborhoods.
Hits & misses are defined uniquely for each neighborhood

To adapt relief to survival data, we can apply a filter to the data to classify instances as hits or misses for each neighborhood where each neighborhood uses its $T(R_i)$ as a reference.
The neighborhood in relief is analogous to one of the partial likelihood functions terms in cox regression.

Bavetta et al. reclassification scheme



Notice that in this classification scheme, even censored R_is can be used whereas in cox regression, they cannot.

Weight update Rule : Relief versus updr

Bavetta et al uses the following weight update rule

for $A := 1$ to A do

$$w[A] := w[A] - \sum_{j=1}^{k(z_i)} Diff(A, R_i, H_j) / k(z_i) \cdot m + \sum_{j=1}^{k(z_i)} Diff(A, R_i, M_j) / k(z_i) \cdot m$$

where hits are instances which share a prototype as determined by the filtering scheme detailed earlier

∴ This is simply the rowsum of the design matrix

∴ convert $\Delta_{pheno} = 0$ (hits) to -1

then $W[A] = \text{rowsum}\left(\Delta_{pheno} \cdot \Delta_{att}\right)$

weighting neighbor distances:

Barreta et al.

$$\text{Diff}\left(A, R_i, R_i^-\right) = \text{Diff}\left(A, R_i, R_i^-\right) / \breve{S}\left(T(R_i)\right)$$

we want to increase attribute differences for instances which have different survival time and outcome (this also helps to account for information loss due to censoring)

I have chosen to make this effect exponential. Neighbors in attribute space are exponentially further away if they have different survival estimates

$$K_i = \exp\left(\frac{S(\breve{T}(R_i)) - S(\breve{T}(NN_i))}{2\sigma^2}\right)$$

↑ user input

Then

$$\Delta_{att_i} = \Delta_{att_i} \cdot K_i$$