Modeling Mortgage Defaults - Part 3: Cox Regression and Hazard Rate Models

In Parts 1 and 2 we estimated a logistic model, and a logistic-transition model on mortgage default data. We saw that the logistic model is a one-period model. The logistic-transition model extended the framework to multiple periods. Each period was allowed to have a different covariate loadings allowing it to capture the varying impact of say FICO on default probability in each discrete period.

The Cox regression model is one of the primary models used in survival analysis. It lets us incorporate time into the model (but time in an ordinal way, more this below). It has a very flexible framework and can accomodate a host of other modeling accoutrements.

Model Summary

- Cox regression extends the idea of 'the odds of something happening' (a one-period concept) to something that happens over time, a rate. The hazard rate is the probabilty of failure given no failure to that point. It's the instantaneous probability of failure, or in our case of discrete time intervals, it is the probability of failure over the next period given no failure to that point.
- Things start by defining the functional form of the hazard rate, for observation k as:

$$h_k(t, X) = h_0(t) x e^{\sum_i \beta_i X_{i,k}}$$

The vanilla hazard rate model defines a hazard rate that is the product of a baseline hazard curve $h_0(t)$, that is a function of time. It is helpful to think of this as the hazard rate for an average loan and does not explicitly involve the covariates (we can think of their effect as baked into this baseline hazard). This is multiplied by $e^{\sum_i \beta_i X_{i,k}}$, a function that varies with the covariates. We can think of this component as adding the observation-specific variation to each observation's hazard rate.

The base formulation of the model is a 'proportional hazards' model, and we can see that if two observations' characteristics (ie. their covariates) are unchanging over time, their ratio of their hazards will be constant, or proportional. This is an assumption of the base model, that hazards be proportional (and this can be tested), however the model can also be generalized to include estimated with non-proportional hazards. In fact, nothing about the estimation method really needs to change in order to do this. An example of non-proportional hazards in our context would be adding monthly home price changes to our model.

Extensions to the base Cox proportional hazards regression include:

- time-varying covariates - we could include monthly HPI values for each loan;
- time-varying coefficients - we could explore the changes in predictive power of FICO, how helpful is it in the first two years versus thereafter;
- adjustments for unexplained heterogeneity, e.x. missing variables - frailty models allow for idiosyncratic dynamics;
- multi-state hazard rates (including recurrent events) - what if there are multiple states an observation can enter? Competing risks is an example of this type of problem (we explore this below in the context of prepay and default);

Model Estimation

There are two components to the estimation process, and R easily does both. The first is to estimate the loadings of the covariates (in the base formulation these are the values that are proportional), this is done by maximum likelihood estimation, where the likelihood is given by a ratio of the hazards of the observations that fail in a given period over the hazards of the 'at risk' (ie. alive) population that started the period.

The likelihood function, L_j, for failure time, j, is the ratio of hazards,

$$L_j = \frac{h_i(j, X)}{\sum_{k=AtRisk_j} h_k(t, X)}$$

The likelihood is then maximized over all failure times, j. In the example above, observation i is the unique failure at time j. The At Risk set at time j is set of all observations 'alive' at that point. We can see that creating the likelihood in this fashion cancels out the baseline hazard curve from each observations hazard rate - the baseline hazard is not therefore not estimated in this process.

The second thing we want is the baseline hazard function itself. There is infact no constraint on the shape of the baseline hazard function itself (ie. non-parametric), and there are several estimators for it (eg. Breslow's estimator). It can be thought of as fitting a function to the data, given we know the loadings of the covariates from the step above. The output from R is a step-function of values for the hazard rate observed during each interval (the intervals do not need to be equally spaced, more on this later).

Hazard Rates and the Survival Function: A Fundamental Relationship

An important set of relationships are those between the hazard rate, the cumulative hazard, and the survival function, ie. the remaining population. The cumulative hazard is the integral of the hazards. In a discrete context, it is the sum of the hazards for each period. The hazard rate is related to the survival function by the following formula.

$$\frac{\frac{\delta S(t)}{\delta t}}{S(t)} = -h_k(t, X)$$

where S(t) is the survival function.

Integrating gives us :

$$S(t) = e^{\int_0^t -h_k(t,X)}$$

Competing Risks Structure

Even though we may only be concerned with modeling defaults over time we forced to consider the entire transition space; loans that prepay are not around to default. We saw this problem when modeling the Logistic Transition model in Part 2. Without modeling prepayments we would be assuming the prepayment rate to be zero (obviously an incorrect assumption), thus over-estimating portfolio defaults. In the competing risks framework, models for each transition are estimated seperately. From the perspective of transition model A, transitions to all other states are treated as censorings.

Building Blocks - The Kaplan-Meier Estimator

We start by estimating the Kaplan-Meier Estimator (KME) for the population. The KME is a step-wise function describing the survival probability as a function of time. It has a number of interesting properties but simply put it decreases (proportionately) whenever there is an exit (but not a censoring) from the population. A censoring is an exit from the population due to lack to data. Maybe we stopped tracking this loan, or the data was no longer available. The earlier models could not incorporate censored data since they required complete follow-up of all observations at the end of each modeled period. An important caveat however is that the censorings must be non-informative (see **Key Concept**: Non-Informative Censoring below).

> In a population of 20, if 10 are censored at t=1 and 5 exit at t=2, the KME will be flat at t=1 and decrease to 0.5 at t=2. If 5 exit at t=1 and 10 are censored at t=2, the KME will decrease to 0.75 at t=1, and stay flat through t=2.

Censored observations determine the likelihood of an exit when they're present, but leave no trace on the population upon their exit.

The KME does not accept any covariates. This makes it it helpful as a summary measure rather than a predictive one. It is nevertheless a good place to start. Let's use our familiar training data and plot its KME.

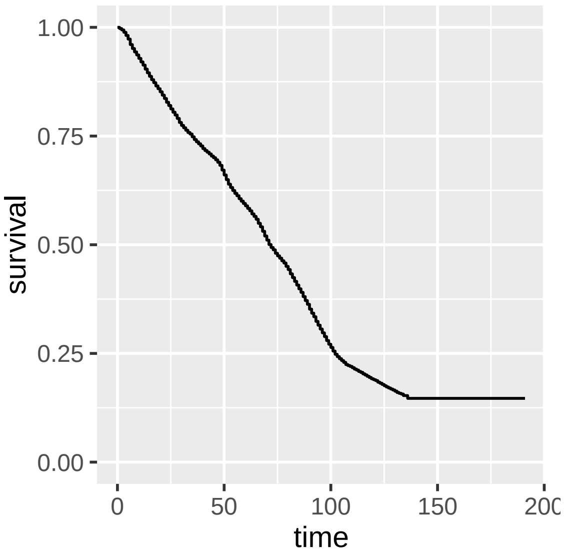
We'll be using the following packages: survival and cmprsk for the surival functions, ggplot and gridExtra for plotting, and dplyr, reshape2, and DataCombine for data manipulation.

Let's take a look at our dataset. We've added a few columns:

- **etime**: the exit time in month (first payment date to default date)
- **ExitCode**: 0 for alive/censor, 1 for prepaid, 2 for defaulted
- **AllExit**: TRUE for ExitCode 1 or 2, FALSE for ExitCode 0

```
##        LOAN_ID   ORIG_DTE OCLTV CSCORE_B   SATO DEFAULT.CODE DEFAULT.DATE
## 1 100027499541 2005-02-01    11      779  0.495            1   2009-05-01
## 2 100149048392 2005-01-01    90      729  0.040            1   2008-02-01
## 3 100191345187 2004-12-01    40      783 -0.250            1   2011-09-01
## 4 100193590893 2005-02-01    80      623  0.620            1   2011-09-01
## 6 100427031672 2005-01-01    56      782  0.165            1   2011-07-01
## 7 100554719899 2005-02-01    73      762 -0.255            1   2005-10-01
##   etime ExitCode AllExit
## 1    50        1    TRUE
## 2    36        1    TRUE
## 3    80        1    TRUE
## 4    78        1    TRUE
## 6    77        1    TRUE
## 7     7        1    TRUE
```

```
KME <- survfit(Surv(time= etime, event = AllExit) ~ 1, data = TrainData)

KMEdf <- data.frame(time = c(0,KME$time), survival = c(1,KME$surv))

ggplot(KMEdf, aes(time, survival)) + geom_step() + expand_limits(y=0)
```

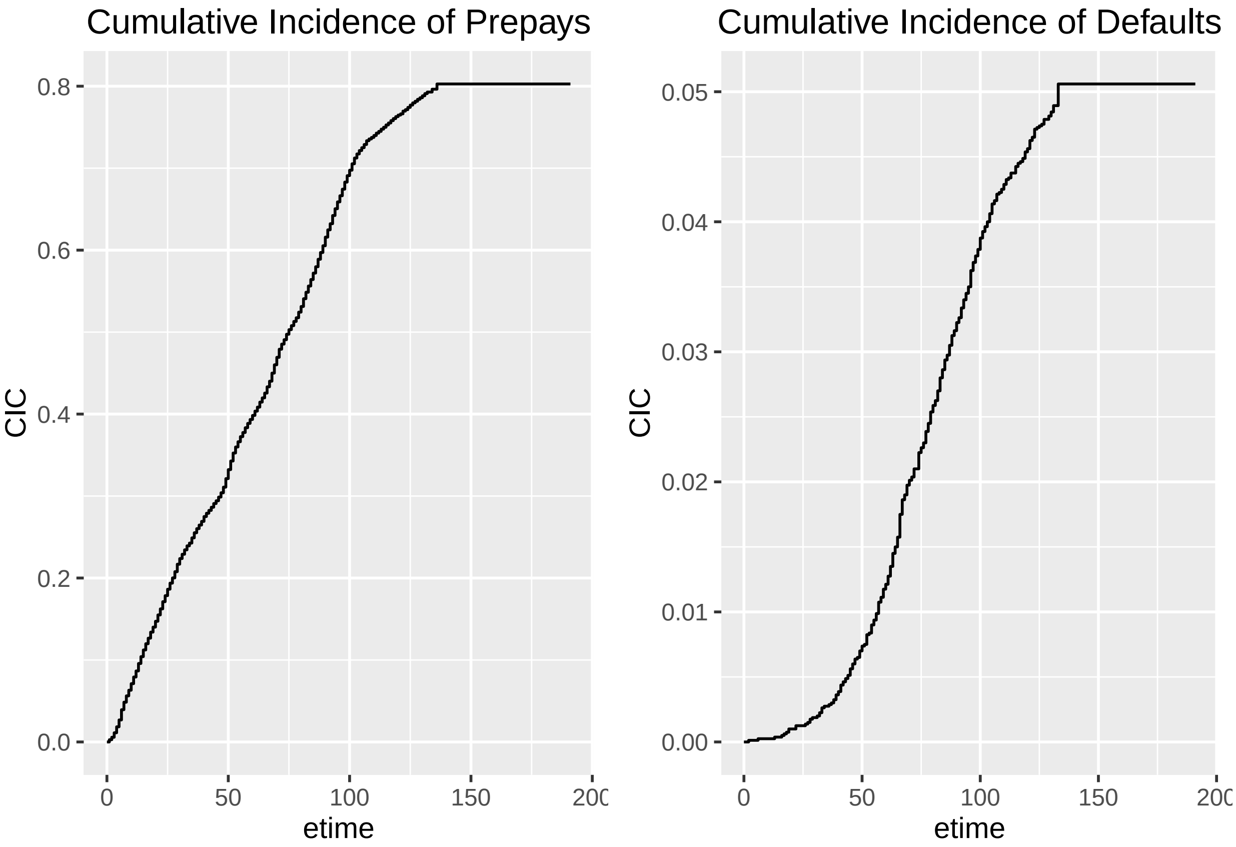


The majority of the 2005-Q1 loans were acquired within several months of origination. For this macroeconomic environment (e.x. rates, house prices, etc.) most of the pool had either prepaid or defaulted by 100 months (~8 years). The median loan surviving for ~60 months. We can see that about 13% of the pool is still current/alive.

Key Concept	A key concept is that of **non-informative censoring**. This means that we should not be able to glean information about the ultimate outcome of a loan from the knowledge that it was censored. For example, if all loan records from a particular issuer were unavailable because they were lost after the originator went bankrupt after being sued when all his loans defaulted, knowledge that they loans were censored **is** informative, since the data is not available because the loans defaulted. If on the other hand records of select loans were unavailable because a computer harddisk was corrupted (a random event, and not linked to the performance of the loans), then we don't necessarily know anything more about the performance of the loans. This would be an example of non-informative censoring.

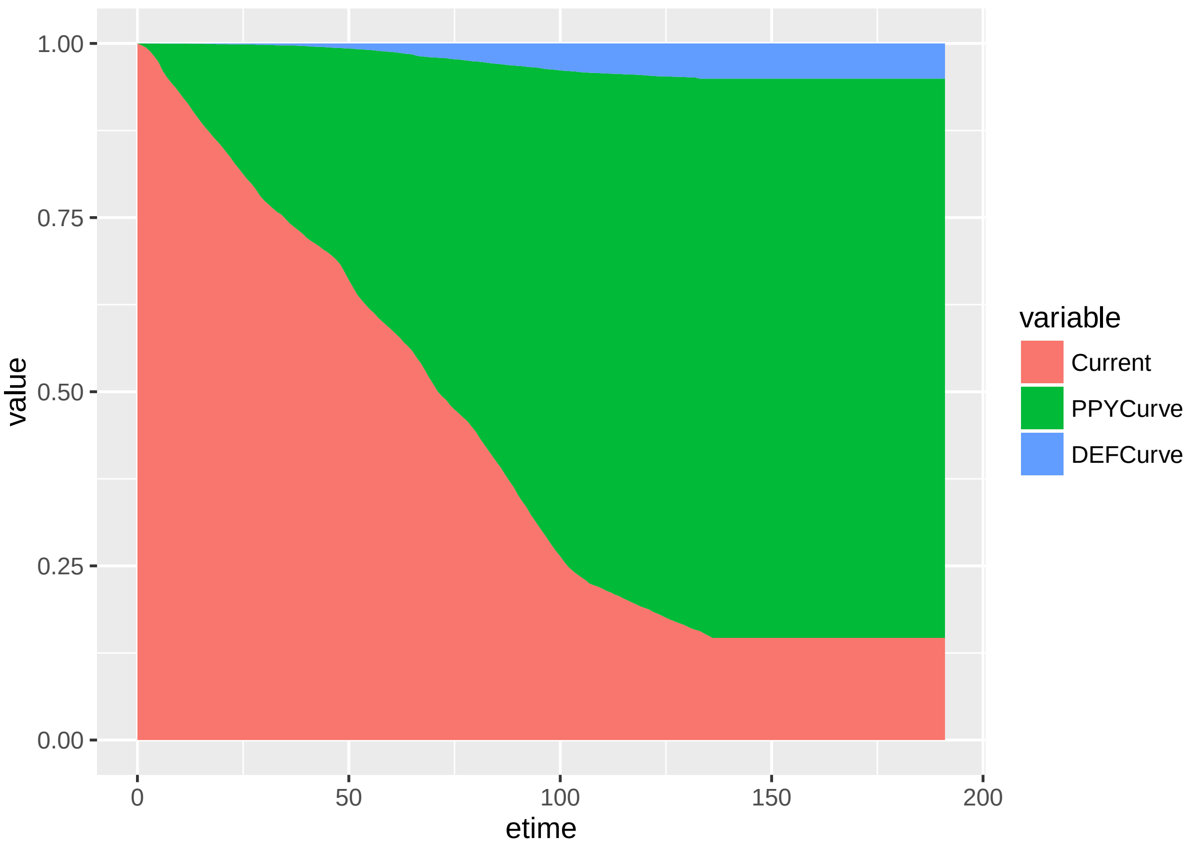
Let's plot the data to visualize what competing risks means. From there we can build a model that predicts the interactions of these risks. For now we stay in data-summary mode. We start with cumulative incidence curve (CIC). These curves attribute population exits by failure type, they are the complement of the KME. Similarly, they address censoring in a consistent way. In the paradigm of the CIC for risk A, exit due to risk B is a censoring, and vice-versa. Recall the importance of non-informative censoring, if the occurence of risk B does tell us something about risk A we can't simply ignore it, which is what we do here.

```
CIC <- cuminc(TrainData$etime, TrainData$ExitCode)

CICdf <- data.frame(ExitReason = factor(rep(c("Prepay", "Default"),
    c(length(CIC[[1]]$time), length(CIC[[2]]$time)))), etime = c(CIC[[1]]$time,
    CIC[[2]]$time), CIC = c(CIC[[1]]$est, CIC[[2]]$est))

g1 <- ggplot(filter(CICdf, ExitReason == "Prepay"), aes(etime,
    CIC)) + geom_step() + ggtitle("Cumulative Incidence of Prepays")
g2 <- ggplot(filter(CICdf, ExitReason == "Default"), aes(etime,
    CIC)) + geom_step() + ggtitle("Cumulative Incidence of Defaults")

arrangeGrob(g1, g2, ncol = 2) %>% plot()
```



It is helpful to see these curves stacked in an area chart

```
PPY <- aggregate(CIC ~ etime, filter(CICdf, ExitReason == "Prepay"),
    FUN = max)
DEF <- aggregate(CIC ~ etime, filter(CICdf, ExitReason == "Default"),
    FUN = max)

colnames(PPY)[colnames(PPY) == "CIC"] <- "PPYCurve"
colnames(DEF)[colnames(DEF) == "CIC"] <- "DEFCurve"

d <- merge(x = PPY, y = DEF, by = "etime", all = TRUE) %>% FillDown("DEFCurve") %>%
    FillDown("PPYCurve") %>% mutate(Current = 1 - PPYCurve -
    DEFCurve)

d <- d[c(1, 4, 2, 3)]

melt(d, id.vars = "etime") %>% ggplot(aes(x = etime, y = value,
    group = variable, fill = variable)) + geom_area(position = "fill")
```



This chart is a summary of our dataset. It is a very useful way of looking at the data. It is however only a summary, and not predictive.

Sidebar: Time Treament in Hazard Rate Models A nuance of the model is that it treats time in an ordinal rather than cardinal manner. The model estimates transitions at each of the event times (exit or censor) in the dataset. The fact that event $t_{failure1} = 1 < t_{failure2} = 5 < t_{failure3} = 10$ is important because the 'at risk' populations are different at each time, but nowhere do we actually use in our modeling the fact that $t_{failure2} = 5$, or whatever the case may be. Certainly we can incorporate 'actual' time into the model as a covariate, but the baseline for each failure time in the model is calculated separately (this goes to the power of the non-parametric, step-wise form). More specifically, if $t_{failure2}$ was 6 instead we would get the same estimates from the cox model for all hazard rates at $t_{failurei}$, assuming the event order was preserved. If we were fitting a spline to the baseline hazard then, yes, the relative spacings of event times would be important. To build intuition for this look at the likelihood function, only the ordering (insofar as it alters the at risk population) matters.

We now estimate the model using variables we earlier established as significant. Normally we'd apply a similar approach to Part 1 for variables selection, but for simplicity we assume these variables are appropriate. We could truncate the data at 6 years as in Parts 1 & 2, but it is helpful to see parallels to the KME, so we run the model to the last loan event. We exclude HPI since it is best incorporated as a time-varying covariate in this context.

We start by estimating the overall hazard exit rate, e.g. prepay and default:

```r
coxExit <- coxph(Surv(etime, ExitCode > 0) ~ OCLTV + CSCORE_B +
    SATO, data = TrainData)
```

```
coxExitCoef <- data.frame(coef(summary(coxExit))) %>% mutate(Change = c(10,
    25, 0.25), ChangeRelativePeriodRisk = exp(coef * Change))

rownames(coxExitCoef) <- rownames(coef(summary(coxExit)))
coxExitCoef[, c(1, 6, 7)]
```

```
##                  coef Change ChangeRelativePeriodRisk
## OCLTV    -0.001841142  10.00                 0.981757
## CSCORE_B  0.001058373  25.00                 1.026812
## SATO      0.123547673   0.25                 1.031369
```

The hazard, or risk, of exiting the pool each period increases by 3.1% for every 0.25% increase in SATO, increases by 2.7% for every 25 point increase in FICO, and (interestingly!) decreases by 1.8% for every 10% increase in OCLTV. Some of the results seem a little odd, going in unexpected directions. What the model is trying to pick up at the same time are both propensities to both prepay and default (good loans tend to prepay and bad ones default). Since there are many more loans that have prepaid than defaulted where prepay and default betas go in different directions it seems reasonable to expect that prepay will win. The impact of these competing risks becomes clearer when we estimte two seperate models.

```
coxPrepay <- coxph(Surv(etime, ExitCode == 1) ~ OCLTV + CSCORE_B +
    SATO, data = TrainData)
coxDefault <- coxph(Surv(etime, ExitCode == 2) ~ OCLTV + CSCORE_B +
    SATO, data = TrainData)
```

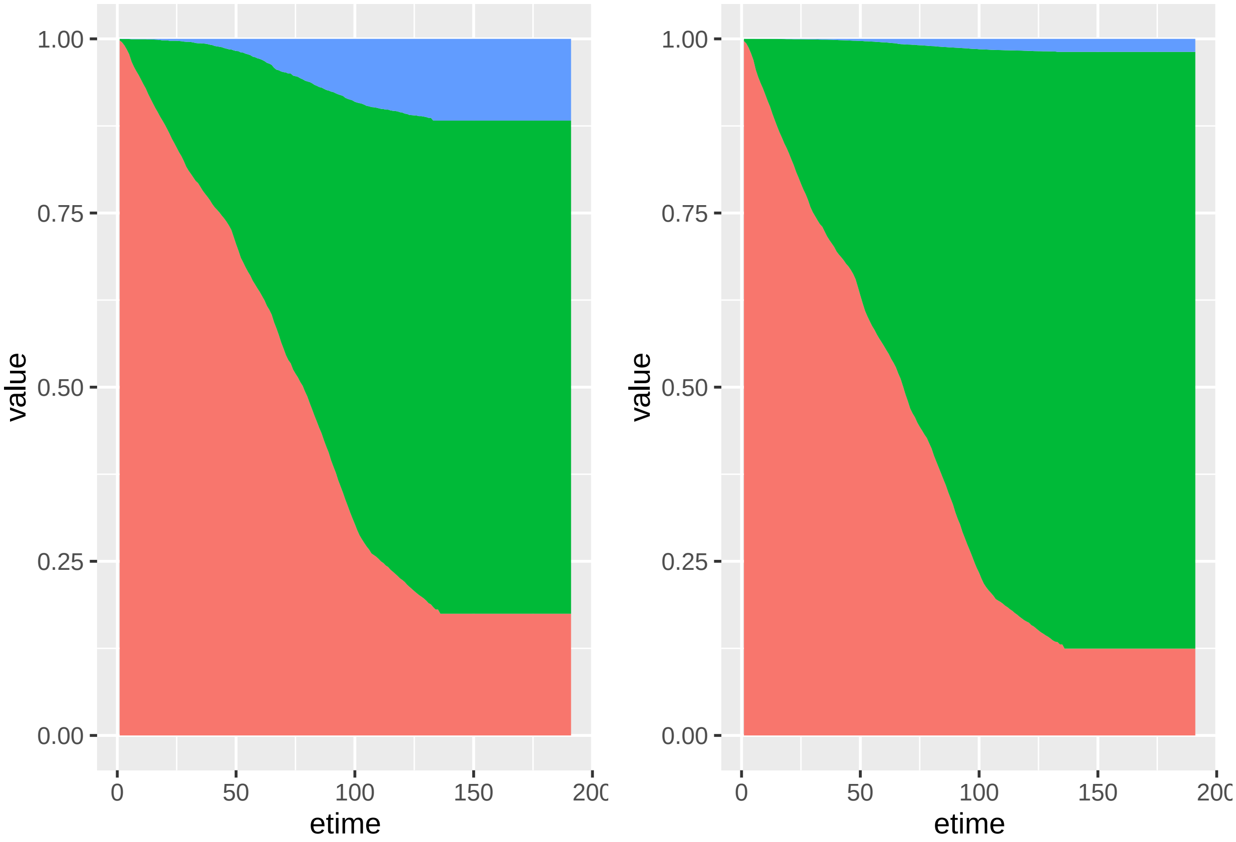
We create a function to compose the two risks. The function works as follows:

1. For each event period we calculate the marginal hazard rates for all candidates and all transitions;
2. We calculate the survival function by applying the marginal hazards of each period to the population remaining at the beginning of each period. Here we are relying on the property that the overall hazard is the sum of the individual hazards, this makes sense since our failure events are mutually exclusive and absorbing. This process of composing the functions is almost identical to chaining of logistic regressions we did in Part 2.
3. To calculate the cumulative incidence curve of each risk we sum the amount of the population that exited the pool each period due to that risk

```
CompetingRiskModel <- function(data, coxModel1, coxModel2) {

    # 1
    SurvivalFunction1 <- survfit(coxModel1, newdata = data)$surv
    SurvivalFunction2 <- survfit(coxModel2, newdata = data)$surv

    CumHaz1 <- -log(SurvivalFunction1)
    CumHaz2 <- -log(SurvivalFunction2)

    MarginalHaz1 <- diff(rbind(0, CumHaz1))
    MarginalHaz2 <- diff(rbind(0, CumHaz2))

    # 2
    AggregateSurvival <- apply((1 - MarginalHaz1 - MarginalHaz2),
        2, cumprod)

    # we want to store event times in the row name
    rownames(AggregateSurvival) <- survfit(coxModel1, newdata = data)$time

    # 3
    n <- nrow(AggregateSurvival)
    CIC1 <- apply(rbind(1, AggregateSurvival[-n, ]) * MarginalHaz1,
        2, cumsum)
    CIC2 <- apply(rbind(1, AggregateSurvival[-n, ]) * MarginalHaz2,
        2, cumsum)

    invisible(list(Prepay = CIC1, Default = CIC2, AggSurv = AggregateSurvival))
}
```

We now have all the pieces to look at some output. Let's compare the difference forecasts for two otherwise identical loans, one with a FICO of 600 (left), and the other of 800 (right).

We note that the curve flattens after month 150. Indeed with only have 2 datapoints past 150 months. There are numerous current loans with approximately 140 years of age, but few current loans beyond (… and certainly no loans defaulting past then, or our survival curve based on only those few loans would have dropped towards zero).



We can see in these graphs the higher FICO loan has a much lower default and a much higher propensity to prepay. The graphs are related proportionately (ie. the cumulative prepay and default hazards for the 600 and 800 FICO loans are proportional to each other) because we do not have, for example, any time-varying covariates.

It is helpful to step back and look at the output. The model as estimated can now calculate the survival probability and cause-specific exit probability, as a function of time, for any loan. The graphs are a visual and intuitive representation of the estimated model.

For readability we skirted many crucial topics. We touch on them briefly here –

Model Assumptions and Assessing Fit

- The hazard model is very flexible and does not impose many restrictions.
- There are many tests of goodness of fit such as:
- the Wald test;
- the Logrank test

These significance tests are important though sometimes limited, warning of poor fits but not more than that.

- Plots of hazard proportionality can be useful in for determining where and whether to use different loadings for a variable over time (e.g. the predictive power of FICO may be different over the first two years than over the remaining life);
- Out of sample, and boundary-value testing will be the most instructive and intuitive measure of model fit;
- non-informative censoring

The baseline hazard curves (estimated as step-wise functions on discrete interval data) can be smoothed using splines. Boundary values may sometimes need to be adjusted ex-post. Towards the end of available data, often where values are sparse, it is not uncommon to have the survival function go negative after composing the independent hazard rates. Of course if you look at the data this does not happen, but towards the endpoints each hazard rate can (estimated and applied ignoring the other exit transitions) can **together** over-allocate the remaining population. Boundary conditions can be imposed to sporadic occurrence correct this.

With the added time dimension there is even more room for creativity in selecting covariates. In the mortgage case we can explore the importance of:

- the macroeconomic environment on borrower prepay/default;
- the current LTV (the amount of equity the borrower has put into the loan over time)
- updated home prices
- updated propensity to refinance / availability of credit (by comparing SATO to prevailing mortgage rates)
- loan age / inertia in decision-making - the incentives and rationale for defaulting/prepaying can change over time, and time-varying covariates can capture this

The hazard rate model is a powerful tool with extensions to model almost any type of survival problem.