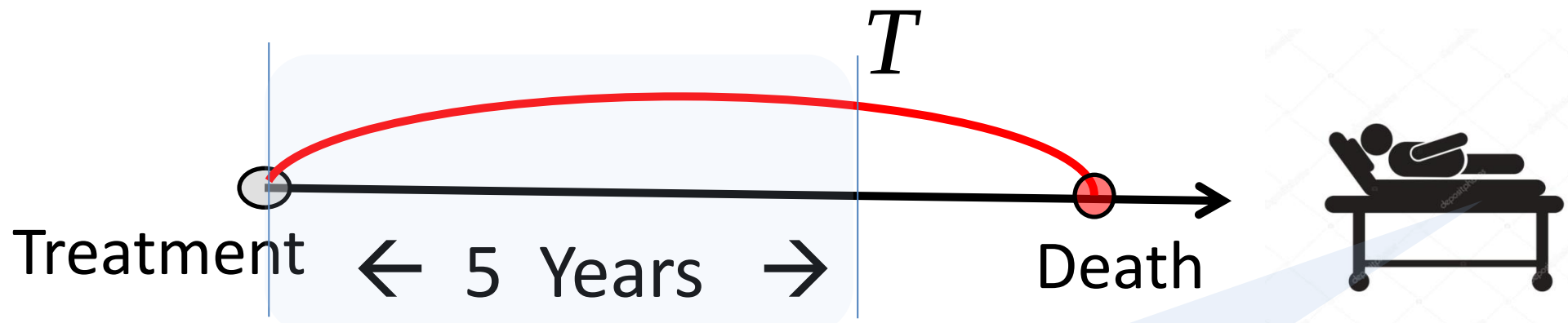
Classification and prediction of survival with high-dimensional features

Takeshi Emura
Graduate Institute of Statistics, NCU

March 8, 2019
Seminar at 國立政治大學統計學系

T = Survival time of a cancer patient



- Features: $\mathbf{x}$ = (Age, Sex, Tumour size, Gene, Cancer Stage, etc.)

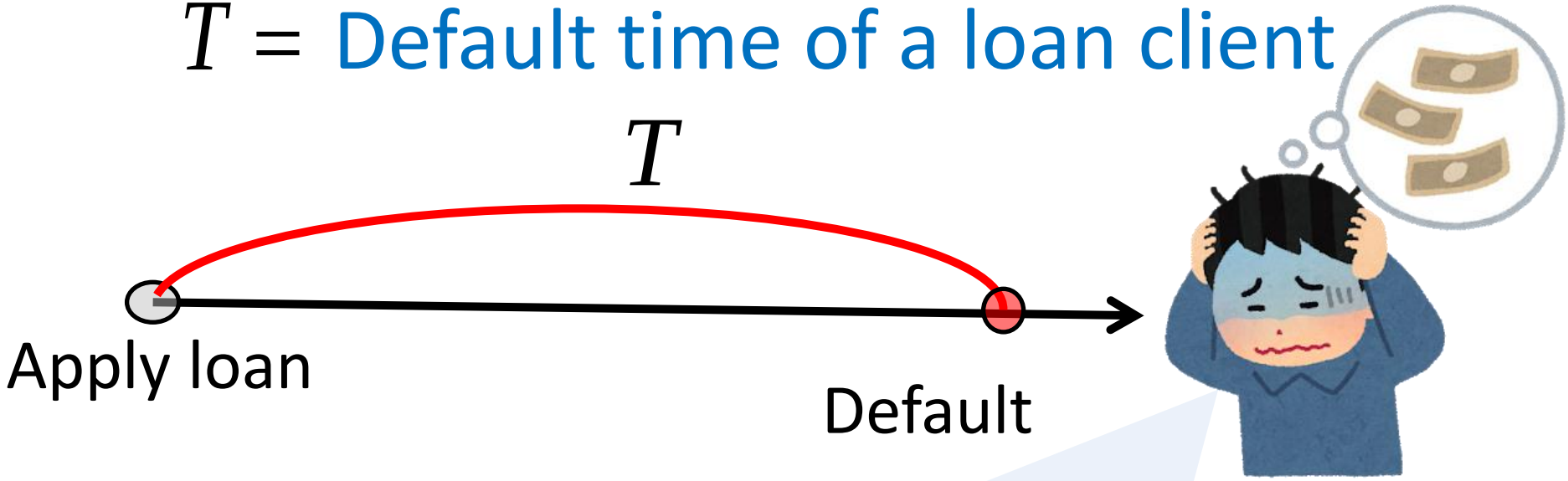
1) Classificiation by $\boldsymbol{\beta}'\mathbf{x}$: Linear Predictor

Short survival (High $\boldsymbol{\beta}'\mathbf{x}$) vs. Long survival (Low $\boldsymbol{\beta}'\mathbf{x}$)

2) Prediction of *t*-year survival probability

$$S(t \mid \mathbf{x}) = \Pr(T > t \mid \mathbf{x})$$

$T =$ Default time of a loan client



- Features: $\mathbf{x}$ =（Gender, age, education, housing status, etc.）

1）Classificiation by $\boldsymbol{\beta}'\mathbf{x}$: Linear Predictor

Risky client（High $\boldsymbol{\beta}'\mathbf{x}$）vs. Safe client (Low $\boldsymbol{\beta}'\mathbf{x}$)

2）Prediction of default probability at time *t*

$$\Pr(T \le t \mid \mathbf{x}) = 1 - S(t \mid \mathbf{x})$$

If linear regression ***were*** applicable…

- Linear model: $T = \mathbf{x}'\boldsymbol{\beta} + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2)$
- Least square estimator: $\hat{\boldsymbol{\beta}} = (X^{\mathrm{T}} X)^{-1} X^{\mathrm{T}} \mathbf{T}$
- Linear predictor and classifier:

$$\hat{y} = \mathbf{x}'\hat{\boldsymbol{\beta}} \qquad \hat{y} \le c \quad \text{vs.} \quad \hat{y} > c$$

- Cross-validation to evaluate prediction error
  ➔ Allen's PRESS (Allen 1974, Technometrics)

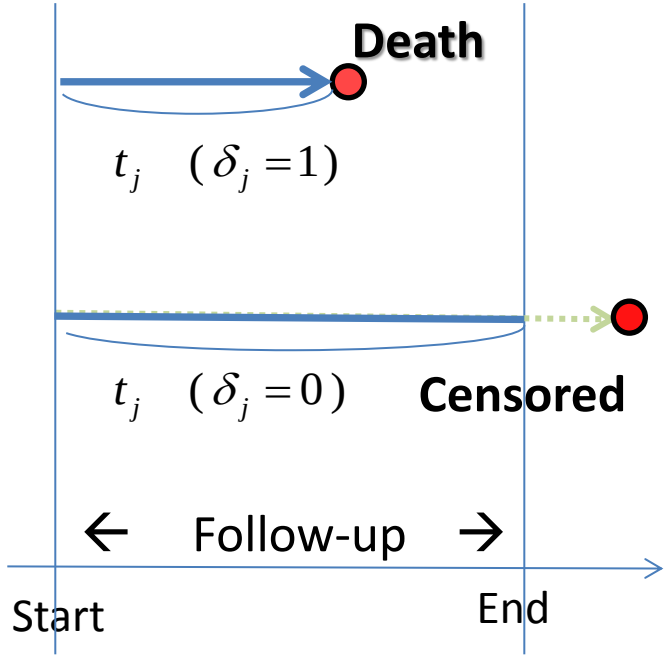
$$\text{PRESS} = \sum_{i=1}^{n} (y_i - \mathbf{x}'\hat{\boldsymbol{\beta}}_{(-i)})^2$$

- Survival Data（right-censored）

t_i : time-to-death or censoring

$$\delta_i = \begin{cases} 1 & \text{if death} \\ 0 & \text{if censoring} \end{cases}$$

$$\mathbf{x}_i = (x_{i1}, ..., x_{ip})', \quad \text{possibly} \quad p > n$$



t_i	δ_i	x_{i1}	x_{i2}	x_{i3}	x_{i4}		x_{i127}	$x_{i,128}$
Time-to-event	Censor	AP3S1	APMAP	ARHGAP28	CXCL12	……..	ASB7	B4GALT5
1650	0	-0.52	1.12	-0.37	1.30	……	0.354	-1.015
30	1	-0.18	-0.69	-0.93	1.28	……	0.026	0.38
⋮	⋮					……		
1800	1	-1.08	0.70	-0.29	-0.529	……	-0.50	-1.09

Short Survival

High Expression

P-value<0.05, but….

Type I error in multiple tests

- **α=0.05 is the error rate of ONLY one test**

  100 tests ➔ $0.05 \times 100 = 5$ rejections

  (False Discoveries)

In Statistical process control (Montgomery 2009)

- **3-sigma** rule, ***α*=0.0023**
- set ***α*** to be **ARL=370**

In Design of microarray experiments (Simon 2001)

- ***α*=0.001,** one error in 1000 tests
- Set ***α*** to be **FDR=0.20**

In Summary…

- "death" or "default" may be unobservable
  ➔ Deal with ***censoring***
  Tool: Cox's partial likelihood (Cox 1972)

- Consider a ***large*** number of features
  ➔ Select a subset of ***features***
  Tool: Multiple significance tests
  Tool: False discovery rate (Witten & Tibshirani 2010)

- Consider a predictor
  Tool: Tukey's compound covariate
  (Tukey 1993; Matsui 2006; Emura et al. 2012, 2019)

- Prediction error via ***cross-validation***
  Tool: Cross-validated likelihood (CVL) (Matsui 2006)

We propose "*compound.Cox*" package

Emura T, Matsui S, Chen HY (2019),
Computer Methods and Programs in Biomedicine
Volume 168: 21-37
https://doi.org/10.1016/j.cmpb.2018.10.020

Tools

- **Lung cancer data**
- **Feature selection**
- **Multiple tests**
- **Predictor calculation**
- **Prediction error (CVL)**
- **False discovery rate (FDR)**
- **Copula methods**

Example: Lung cancer data in *Lung* object

Training samples（n＝63）
Test samples（n＝62）

```
> library(compound.Cox)
> data("Lung")
> Lung
```

↓*p*=97個Features（離散値）

	t .vec	d.vec	train	VHL	IHPK1	…	RPL5
1	47.06271	0	FALSE	2	2		4
2	49.27393	0	TRUE	3	4		4
3	20.06601	1	TRUE	2	3		1
4	26.99670	1	TRUE	2	4		2
5	39.90099	0	FALSE	3	4		4
⋮	⋮	⋮	⋮	⋮	⋮	⋮	
125	56.84141	0	FALSE	3	2	…	3

Survival time (月)

Censor

The data made available in R package *“compound.Cox”* (Emura et al. 2019)

Univariate Cox regression

T = Survival time

x_j = j-th feature

Proportional hazards model (Cox 1972)

$$\Pr(t \le T < t + dt \mid T \ge t, x_j) = h_0(t) \exp(\beta_j x_j) dt$$

Partial likelihood estimator (Cox regression)

$$\hat{\beta}_j = \arg\max \ell_j(\beta_j)$$

$$\ell_j(\beta_j) = \sum_{i=1}^{n} \delta_i \left[\beta_j x_{ij} - \log \left(\sum_{\ell \in R_i} \exp(\beta_j x_{\ell j}) \right) \right]$$

$$SE(\beta_j) = [-\partial^2 \ell_j(\beta_j) / \partial \beta_j^2]^{-1/2}$$

Univariate significance tests

$$H_{oj}: \beta_j = 0 \quad \text{vs.} \quad H_{1j}: \beta_j \neq 0$$

(1) Wald test:

Z-value: $z_j = \hat{\beta}_j / SE(\hat{\beta}_j) \sim N(0,1)$

P-value: $P_j = \Pr(|Z| > |z_j|)$

(2) Score test:

$$S_j = \sum_{i=1}^{n} \delta_i \left(x_{ij} - \bar{x}_j(t_i) \right) \qquad V_j = Var(S_j) = \sum_{i=1}^{n} \delta_i \left(\bar{x}_j^2(t_i) - (\bar{x}_{ij}(t_i))^2 \right)$$

Z-value: $z_j = S_j / \sqrt{V_j} \sim N(0,1)$

P-value: $P_j = \Pr(|Z| > |z_j|)$

-Simple algebraic computation

Multiple score tests via matrix algebras

- Z-values: $\mathbf{Z} = \mathbf{S} / \mathbf{V}^{1/2}$

$$\mathbf{S} = \boldsymbol{\delta}'(\mathbf{X} - \mathbf{S}^{(1)} / \mathbf{S}^{(0)})$$

$$\mathbf{V} = \boldsymbol{\delta}'(\mathbf{S}^{(2)} / \mathbf{S}^{(0)} - (\mathbf{S}^{(1)} / \mathbf{S}^{(0)})^2)$$

$$\boldsymbol{\delta} = (\, \delta_1, \ldots, \delta_n \,)'$$

$\mathbf{S}^{(k)}$ is a $n \times p$ matrix with elements $S_{ij}^{(k)}$

$$S_{ij}^{(k)} = \sum_{\ell : t_\ell \geq t_i} x_{\ell j}^k \quad \text{for} \quad k = 0,\ 1 \text{ or } 2, \text{ and} \quad j = 1, \ldots,\ p$$

This computing technique is **new**
and implemented in *compound.Cox* package

Whole algorithms

Step 1: Test $H_{oj}: \beta_j = 0$ vs. $H_{1j}: \beta_j \neq 0$

Select a feature (via P-value < 0.05 in Wald or Score)

Step 2 : Critical evaluation of selected features

（1）False Discovery Rate (FDR) small enough ?

（2）Cross-validated Likelihood (CVL) high enough ?

Step 3 : Compound covariate predictor

Z-value: $z_1 x_1 + z_2 x_2 + \cdots + z_p x_q$

ß-value: $\hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \cdots + \hat{\beta}_q x_q$

Step 4 : Test the predictor by independent test samples

"*compound.Cox*" (Emura et al. 2019)
is designed to perform the whole algorithms

R函數 *uni.selection( )*

輸入 features（$n \times p$ matrix）

輸入 Survival Time

輸入 Censor

輸入P值

輸入（TRUE or FALSE）

```
> uni.selection(t.vec,d.vec,X.mat,K=20,P.value=0.05,score=FALSE,permutation=TRUE) ## Wald test
$beta
ANXA5       DLG2        ZNF264      DUSP6       CPEB4       LCK         STAT1
-1.0876762  1.3215044   0.5473276   0.7524497   0.5891676   -0.8447389  -0.5844262
RNF4        IRF4        STAT2       HGF         ERBB3       NF1         FRAP1
0.6463635   0.5176704   0.5849869   0.5086750   0.5509026   0.4715235   -0.7696768
MMD         HMMR
0.9151541   0.5156711

$Z
.....  省略
$P
.....  省略

$CVL
-98.66365

$FDR
P.value * (No. of genes)        Permutation
0.3031250                       0.3128125
```

產生回帰係数

$$\hat{\beta}_j = \arg\max \ell(\beta_j)$$

⇐產生Z值

⇐產生P值

⇐產生CVL值

⇐產生FDR值

We selected **16 features** (P<0.05), but…

```
> uni.selection(t.vec,d.vec,X.mat,K=20,P.value=0.05,score=FALSE,permutation=TRUE)
$beta
ANXA5       DLG2       ZNF264      DUSP6      CPEB4       LCK         STAT1
-1.0876762  1.3215044  0.5473276  0.7524497  0.5891676  -0.8447389  -0.5844262
 RNF4        IRF4       STAT2       HGF        ERBB3       NF1         FRAP1
 0.6463635  0.5176704  0.5849869  0.5086750  0.5509026  0.4715235  -0.7696768
 MMD         HMMR
 0.9151541  0.5156711
```

- **Q1: Is there any FALSE feature?**
  ⇒**A1**: Compute FDR（False discovery rate）
- **Q2: Is "P-value < 0.05" optimal selection criterion?**
  ⇒**A2**: Compute CVL（Cross-validated likelihood）

False Discovery Rate (FDR)

FDR= Proportion of false rejection

= $E[f/16]$, where f is unknown

	Rejected	**Accepted**
ß≠0	16-f	
ß=0	f	

q=16 ***p=97***

➔ **FDR** and $E[f]$ can be estimated by permutation method (Witten & Tibshirani 2010)

Random *M* permutations（Witten & Tibshirani 2010）

$$\mathrm{FDR} = \frac{\text{The expected number of false rejections}}{\text{The number of rejections}}$$

$$= \frac{\frac{1}{M}\sum_{m=1}^{M}\sum_{j=1}^{p} I(P_j^{(m)} < P)}{\sum_{j=1}^{p} I(P_j < P)}$$

$P_j^{(m)}$ is the P-value for testing $H_{0j}: \beta_j = 0$ vs. $H_{1j}: \beta_j \neq 0$

- **FDR does not guarantee predictive ability**

  （small FDR  high prediction ability）

CVL（Cross-validated likelihood)

: predictive capability of selected features

$$CVL = \sum_{k=1}^{K} \{ \ell(\hat{\gamma}_{-k}) - \ell_{-k}(\hat{\gamma}_{-k}) \},$$

where $\hat{\gamma}_{-k} = \arg\max_{\gamma} \ell_{-k}(\gamma)$,

CVL= - (PRESS)
for Gaussian likelihood
for uncensored data

Predicted Likelihood ➔ $\ell(\gamma) = \sum_i \delta_i \left[\gamma \mathrm{CC}_{i,-k} - \log \left\{ \sum_{\ell \in R_i} \exp\left(\gamma \mathrm{CC}_{\ell,-k} \right) \right\} \right],$

Baseline ➔ $\ell_{-k}(\gamma) = \sum_{i \in \Im_{-k}} \delta_i \left[\gamma \mathrm{CC}_{i,-k} - \log \left\{ \sum_{\ell \in R_i \cap \Im_{-k}} \exp\left(\gamma \mathrm{CC}_{\ell,-k} \right) \right\} \right],$

High CVL ➡ High prediction (classification) capability

Matsui 2006 ; Emura, Matsui and Chen 2019

$$\begin{bmatrix} t_1, \delta_1, x_{11}, x_{12}, \ldots\ldots\ldots\ldots, x_{1p} \\ \vdots \\ t_n, \delta_n, x_{n1}, x_{n2}, \ldots\ldots\ldots\ldots, x_{np} \end{bmatrix}$$: survival data with n samples

⇓ ***Step 1***: Divide between "testing" vs. "training" samples

$$\begin{bmatrix} t_1, \delta_1, x_{11}, x_{12}, \ldots\ldots\ldots\ldots, x_{1p} \\ \vdots \\ \text{------} \\ \vdots \\ t_n, \delta_n, x_{n1}, x_{n2}, \ldots\ldots\ldots\ldots, x_{np} \end{bmatrix}$$

$\frac{n}{K}$ testing samples, $i \in \mathfrak{I}_k$

$n - \frac{n}{K}$ training samples, $i \in \mathfrak{I}_{-k}$

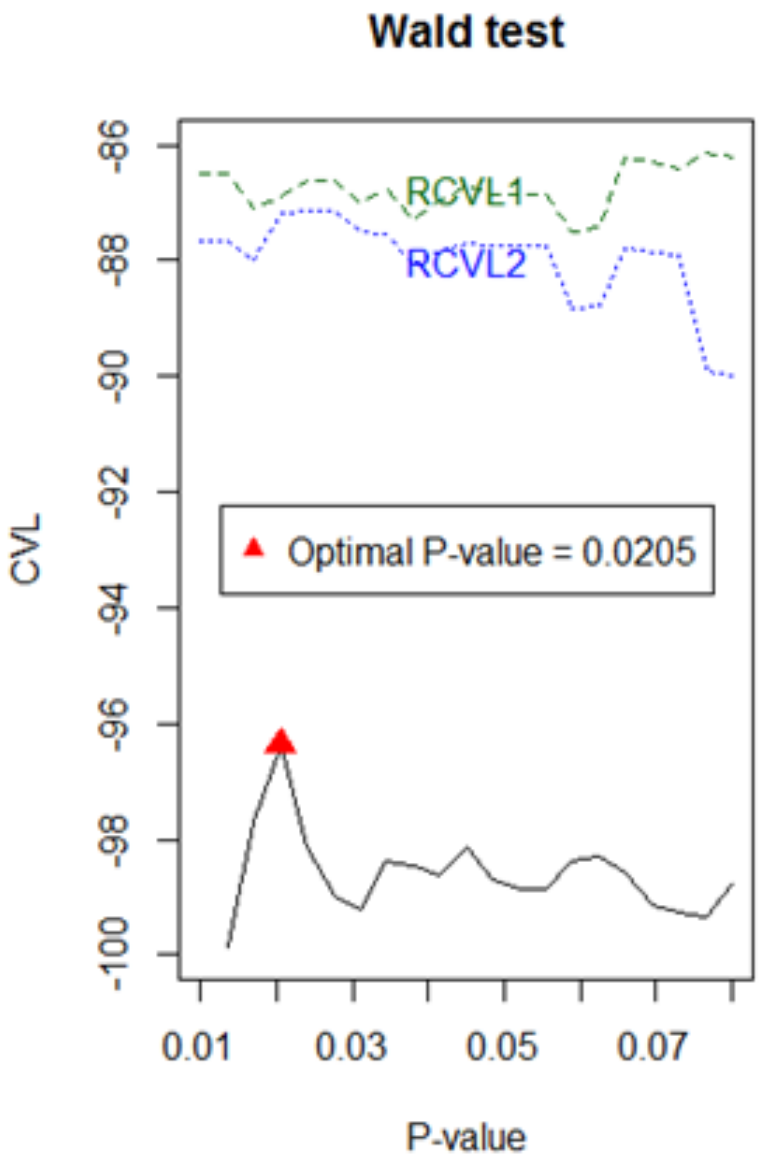
⇓ ***Step 2***: Feature Selection & Prediction (P-values < P)

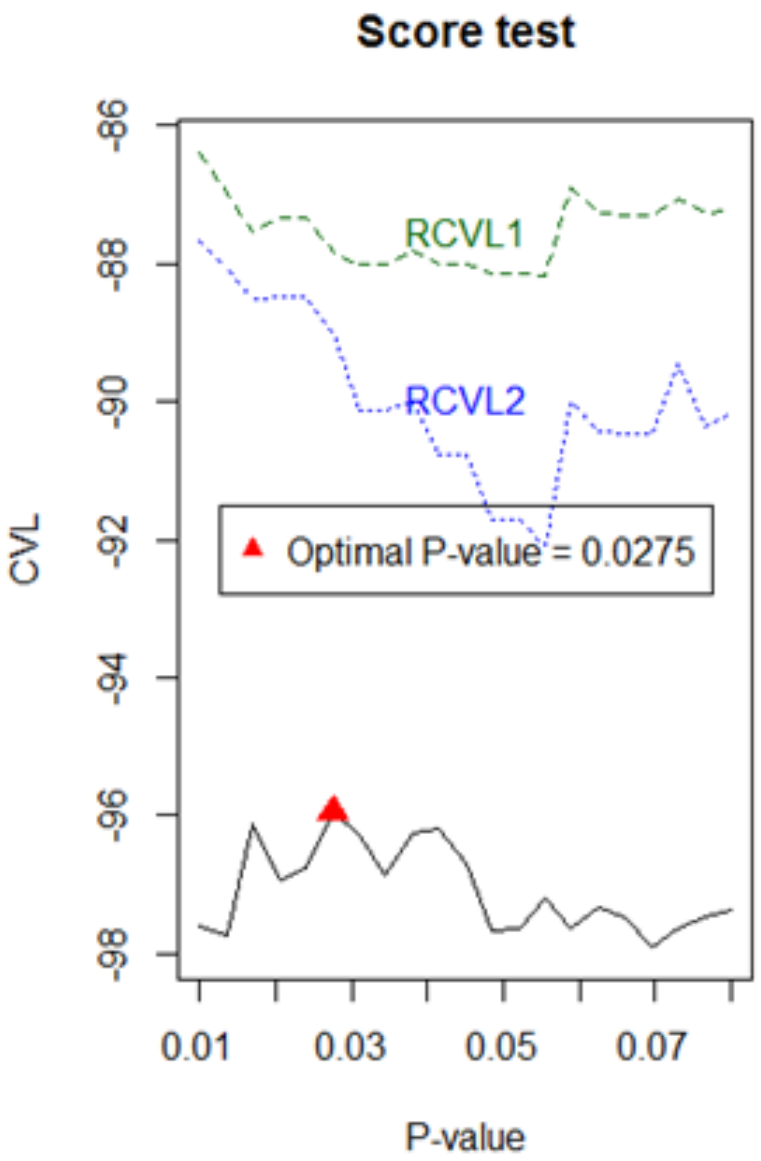
$$\begin{bmatrix} t_1, \delta_1, \mathrm{CC}_{1,-k} \\ \vdots \\ t_m, \delta_m, \mathrm{CC}_{m,-k} \end{bmatrix}, \quad \mathrm{CC}_{i,-k} = w_{1,-k} x_{i1} + w_{2,-k} x_{i2} + \cdots + w_{q,-k} x_{iq}, \quad i \in \mathfrak{I}_{-k}$$

⇓ ***Step 3***: Average (Predicted likelihood - Baseline)

$$CVL = \sum_{k=1}^{K} \{ \ell(\hat{\gamma}_{-k}) - \ell_{-k}(\hat{\gamma}_{-k}) \}$$

The optimal **P-value threshold** by the CVL plot





Optimal Wald tests (P< 0.0205)
➔ Select 7 features (genes)

```
> uni.selection(t.vec,d.vec,X.mat,K=20, P.value=0.0205,score=FALSE)
$beta
ANXA5       DLG2       ZNF264      DUSP6      CPEB4      LCK         STAT1
-1.0876762  1.3215044  0.5473276   0.7524497  0.5891676  -0.8447389  -0.5844262

$CVL  -96.37303
```

↑CVL

FDR=0.0205×97/7=0.29（29%）

Optimal score tests P <0.0275)
➔ Select 10 features (genes)

```
> uni.selection(t.vec,d.vec,X.mat,K=20, P.value=0.0275,score=TRUE)
$Z
ANXA5      DLG2       ZNF264     DUSP6      CPEB4      LCK         STAT1       STAT2
-3.363578  3.111772   2.814363   2.710854   2.538888   -2.511423   -2.445038   2.369334
RNF4       IRF4
2.345912   2.231286
⋮
$CVL    -95.95690
```

↑CVL

FDR=0.0275×97/10=0.30（30%）

Prediction & Classification

- **Selected Features**; $(x_1, \ldots, x_q)$

  In the optimal score tests, $q = 10$

- **Compound Covariate**:

$$\mathbf{CC} = w_1 x_1 + \cdots + w_p x_q$$

$$\beta\text{-weight: } (w_1, \ldots, w_q) = (\hat{\beta}_1, \ldots, \hat{\beta}_q)$$

$$Z\text{-weight: } (w_1, \ldots, w_q) = (z_1, \ldots, z_q)$$

- Classification **CC** < *c* ➡ Good prognosis (Long survival)

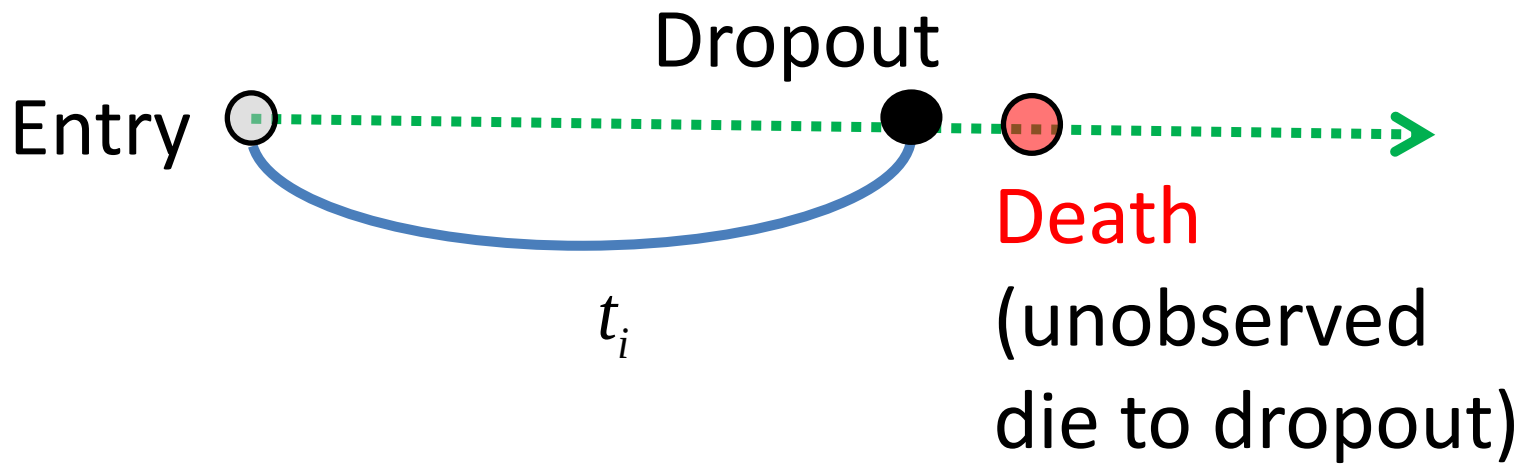
  CC > *c* ➡ Poor prognosis (Short survival)

  ⬆ cut-off value

Bias due to dependent censoring

Many dropouts before death

➡ Dependence exists between censoring and death



Estimate $\hat{\beta}_j = \arg\max \ell_j(\beta_j)$ is biased

(Emura & Chen 2016; 2018)

Treatment of Dependent Censoring

T = Survival (death) time

U = Censoring (dropout) time

x_j = *j-th feature*

C_α = Copula; α = copula parameter

⇓ Joint survival function

$$\Pr(T_i > t, U_i > u \mid x_{ij}) = C_\alpha\{\Pr(T_i > t \mid x_{ij}), \Pr(U_i > u \mid x_{ij})\}$$

$$\Pr(T_i > t \mid x_{ij}) = \exp\{-\Lambda_{0j}(t) e^{\beta_j x_{ij}}\}$$

Effect of j-th feature on T

⇑ Marginal survival function

Estimation under dependent censoring

Semi-parametric MLE (Chen 2010; Emura and Chen 2016)

$$\ell(\beta_j, \gamma_j, \Lambda_{0j}, \Gamma_{0j} \mid \alpha)$$

$$= \sum_i \delta_i [\beta_j x_{ij} + \log \eta_{1ij}(t_i; \beta_j, \gamma_j, \Lambda_{0j}, \Gamma_{0j} \mid \alpha) + \log d\Lambda_{0j}(t_i)]$$

$$+ \sum_i (1-\delta_i)[\gamma_j x_{ij} + \log \eta_{2ij}(t_i; \beta_j, \gamma_j, \Lambda_{0j}, \Gamma_{0j} \mid \alpha) + \log d\Gamma_{0j}(t_i)]$$

$$- \sum_i \Phi_\alpha[\exp\{-\Lambda_{0j}(t_i)e^{\beta_j x_{ij}}\}, \exp\{-\Gamma_{0j}(t_i)e^{\gamma_j x_{ij}}\}],$$

Computed by "*compound.Cox*" R package

$$(\hat{\beta}_j(\alpha), \hat{\gamma}_j(\alpha), \hat{\Lambda}_{0j}(\alpha), \hat{\Gamma}_{0j}(\alpha))$$

Compute weight w_j the CC

Survival prediction

1. **Optimal Wald（7 features）:**

$$\mathbf{CC}= \hat{\beta}_1 x_{i1} + \cdots + \hat{\beta}_7 x_{i7}$$

2. **Optimal score（10 features）:**

$$\mathbf{CC}= z_1 x_{i1} + \cdots + z_{10} x_{i10}$$

3. **Optimal Wald + copula（7 features）:**

$$\mathbf{CC}= \hat{\beta}_1(\hat{\alpha}) x_{i1} + \cdots + \hat{\beta}_7(\hat{\alpha}) x_{i7}$$

4. **Optimal score + copula（10 features ）:**

$$\mathbf{CC}= \hat{\beta}_1(\hat{\alpha}) x_{i1} + \cdots + \hat{\beta}_{10}(\hat{\alpha}) x_{i10}$$

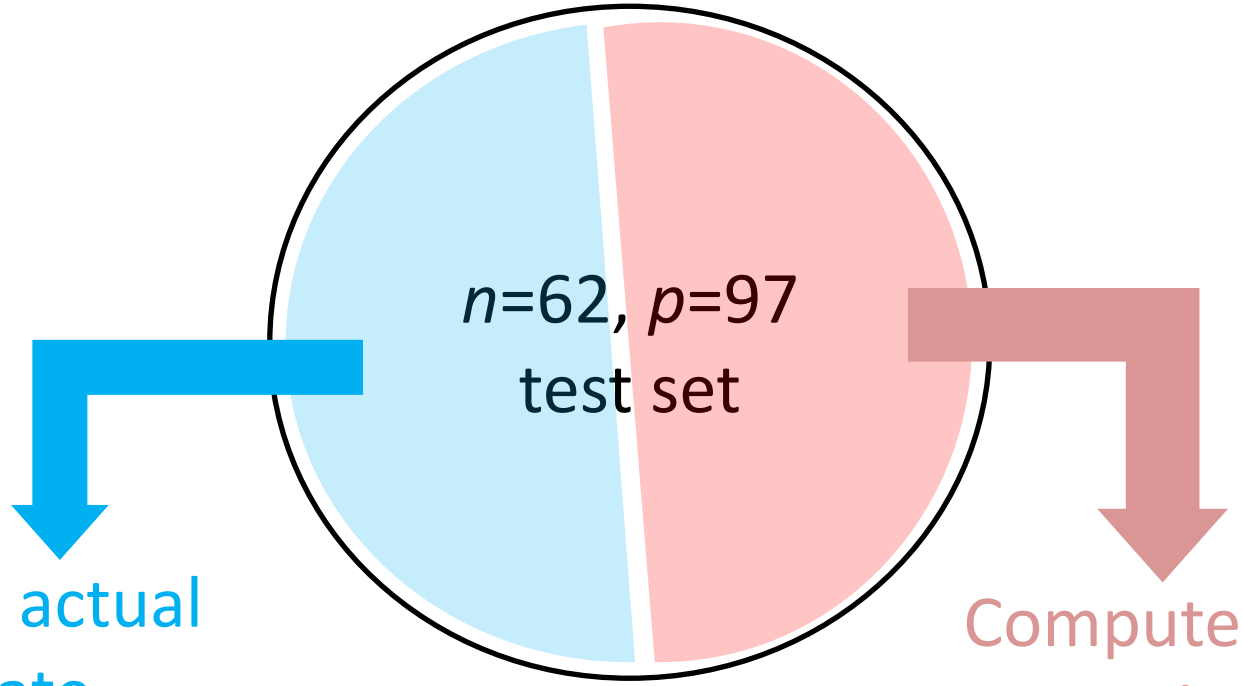
$CC < c$ ➡ Good prognosis（High survival rate）

$CC > c$ ➡ Poor prognosis（Low survival rate）

Classify the **test** samples（$n=62$）

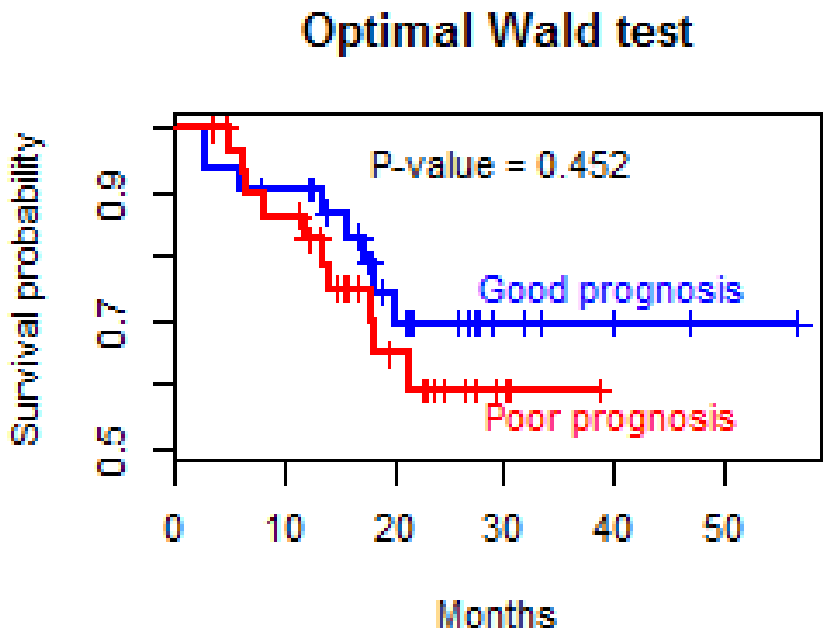
Class 1；Good prognosis（Low CC）

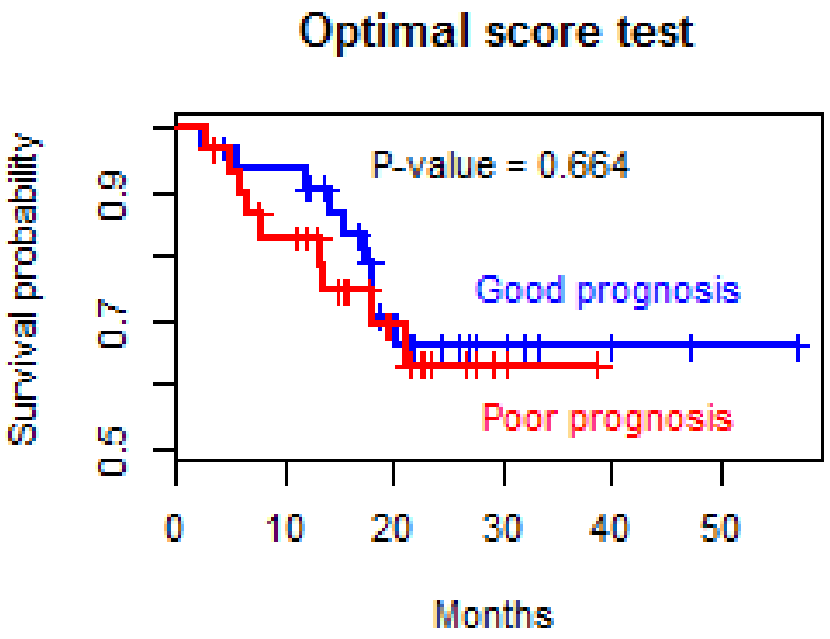
Class 2: Poor prognosis（High CC）

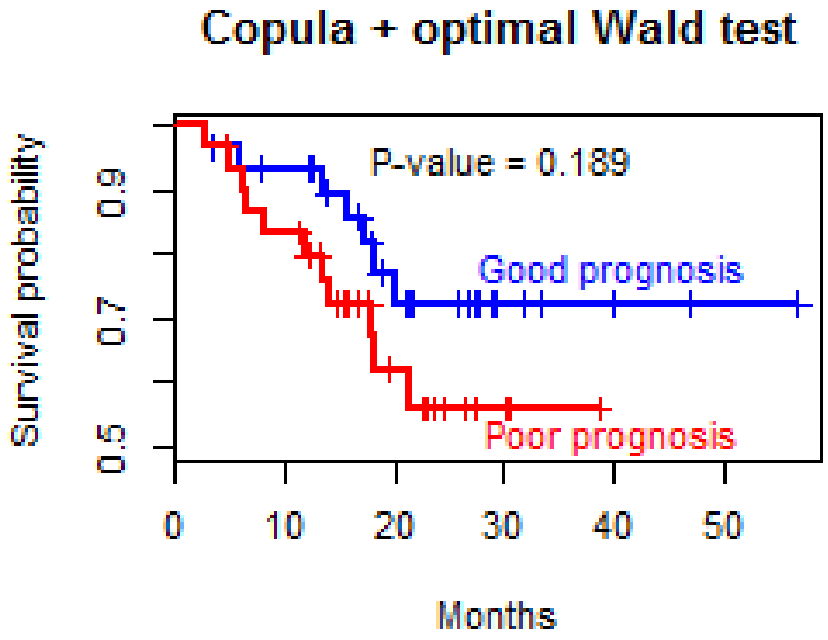


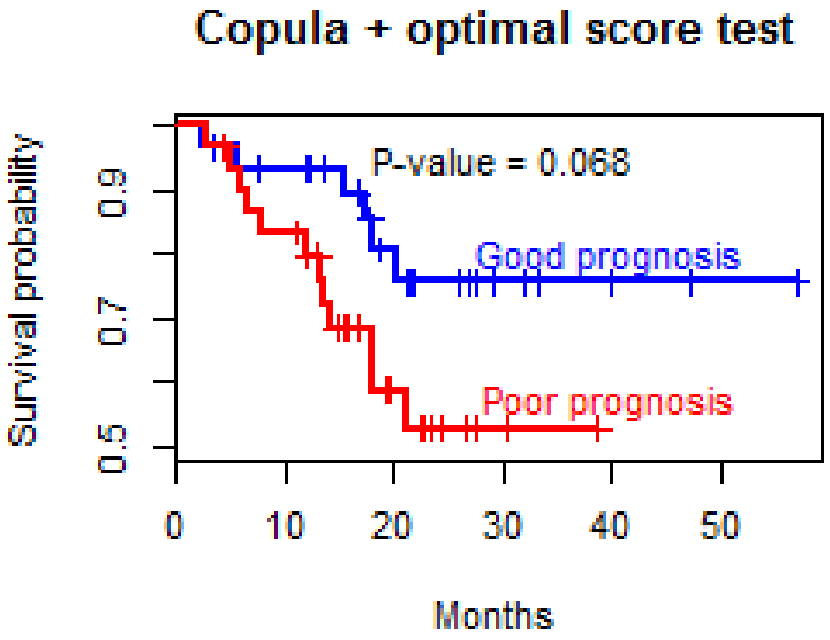
Compute actual survival rate (KM survival curve)

Compute actual survival rate (KM survival curve)


Optimal Wald test
P-value = 0.452
Good prognosis
Poor prognosis
Survival probability
0.5
0.7
0.9
0
10
20
30
40
50
Months


Optimal score test
P-value = 0.664
Good prognosis
Poor prognosis
Survival probability
0.5
0.7
0.9
0
10
20
30
40
50
Months


Copula + optimal Wald test
P-value = 0.189
Good prognosis
Poor prognosis
Survival probability
0.5
0.7
0.9
0
10
20
30
40
50
Months


Copula + optimal score test
P-value = 0.068
Good prognosis
Poor prognosis
Survival probability
0.5
0.7
0.9
0
10
20
30
40
50
Months

Summary

- **Developed an R package "*compound.Cox*"**
  - Use **multiple tests** for feature selection
    (frequently used in medical research)
  - Use **compound covariate** for prediction
    (a direct link to multiple tests)
  - Use a **vector computation** of score tests (new method)
- **Implemented the evaluation measures:**
  - Predictive capability (CVL) Matsui (2006)
  - False discovery rate (FDR) Witten and Tibshirani (2010)
- **Used copula for deal with dependent censoring**
  - More accurate predictor if censoring is informative

Future works

- **Applications to Network Learning**

  - NL requires feature selection in a initial step, but many existing methods treat survival data as “binary” (death vs. survive)”. (Kim et al. 2018)

  - *t*-test → score test (get higher efficiency!)

- **Applications to Finance, Econometrics, Business**

  (1). Analysis of unemployment duration (Lo et al. 2017)

  (2). Time-to-insolvency of a company (Frank and Dörre 2017; Achim and Emura 2019)

  (3). Time-to-default for a loan client (Dirick et al. 2017).

  - High-dimensional features (marriage, job, education, etc.)

Thank you !

References

[1] Achim D, Emura T, Analysis of Doubly Truncated Data, An Introduction, JSS Research Series in Statistics, Springer, Singapore; 2019.

[2] Witten M, Tibshirani R. Survival analysis with high-dimensional covariates. Statist Method Med Res 2010; 19: 29-51.

[3] Matsui S. Predicting survival outcomes using subsets of significant genes in prognostic marker studies with microarrays. BMC Bioinformatics 2006; 7:156.

[4] Chen HY, Yu SL, Chen CH, Chang GC, Chen CY, et al. A five-gene signature and clinical outcome in non-small-cell lung cancer. N Engl J Med 2007; 356: 11-20.

[5] Dirick L, Claeskens G, Baesens B. Time to default in credit scoring using survival analysis: a benchmark study. Journal of the Operational Research Society 2017, 68(6), 652-665.

[6] Emura T, Chen YH, Chen HY. Survival prediction based on compound covariate under Cox proportional hazard models. PLoS One 2012; 7(10): e47627. DOI:10.1371/journal.pone.0047627.

[7] Emura T, Chen YH, Gene selection for survival data under dependent censoring, a copula-based approach, Statist Method Med Res 2016; 25(6): 2840-2857.

[8] Emura T, Chen YH, Analysis of Survival Data with Dependent Censoring, Copula-Based Approaches, JSS Research Series in Statistics, Springer, Singapore; 2018.

[9] Emura T, Matsui S, Rondeau V (2019), Survival Analysis with Correlated Endpoints, Joint Frailty-Copula Models, JSS Research Series in Statistics, Springer, Singapore; 2019

[10] Frank G, Dörre A. Linear regression with randomly double-truncated data. South African Statistical Journal 2017, 51(1), 1-18.

[11] Kim M, Oh I, Ahn J. An improved method for prediction of cancer prognosis by network learning. Genes 2018; 9: 478.

[12] Lo SM, Stephan G, Wilke RA. Competing risks copula models for unemployment duration: An application to a German Hartz reform. Journal of Econometric Methods 2017, 6(1)