

# Histopathology Research Template

*true*

*2019-11-06*

## Contents

<b>1</b>	<b>impute</b>	<b>3</b>
1.1	impute continious . . . . .	3
1.2	impute categorical . . . . .	3
1.3	impute outlier . . . . .	3
<b>2</b>	<b>transform</b>	<b>3</b>
2.1	min -max . . . . .	3
2.2	skewness . . . . .	3
2.3	log . . . . .	3
<b>3</b>	<b>binning</b>	<b>3</b>
3.1	optimal binning . . . . .	3
3.2	standardize . . . . .	3
<b>4</b>	<b>data transformation report</b>	<b>3</b>
<b>5</b>	<b>inspectdf</b>	<b>3</b>
<b>6</b>	<b>Descriptive Statistics</b>	<b>4</b>
6.1	Table 1 . . . . .	4
6.2	Categorical Variables . . . . .	4
6.2.1	Split-Group Stats Categorical . . . . .	4
6.2.2	Grouped Categorical . . . . .	4
6.3	Continious Variables . . . . .	5
6.3.1	Split-Group Stats Continious . . . . .	5
6.3.2	Grouped Continious . . . . .	5
<b>7</b>	<b>Cross Tables</b>	<b>6</b>
<b>8</b>	<b>Plots</b>	<b>7</b>
8.1	Categorical Variables . . . . .	7
<b>9</b>	<b>Plots</b>	<b>7</b>
9.1	Continious Variables . . . . .	7
<b>10</b>	<b>Survival Analysis</b>	<b>8</b>
<b>11</b>	<b>Pairwise comparison</b>	<b>8</b>
<b>12</b>	<b>Multivariate Analysis Survival</b>	<b>8</b>
<b>13</b>	<b>KM plot</b>	<b>8</b>
<b>14</b>	<b>Separate tables</b>	<b>9</b>
<b>15</b>	<b>Pipe together</b>	<b>9</b>

16 Separate tables	10
17 Pipe together	10
18 OR plot	10
19 HR plot (not fully tested)	10
20 ANOVA	10
21 Save Final Data	10
22 Final Data Summary	11
23 Software and Libraries Used	12
24 Session Info	13
25 Notes	14

```
iris %>% mutate(sumVar = rowSums(.[1:4]))
```

```
iris %>% mutate(sumVar = rowSums(select(., contains("Sepal")))) %>% head
```

```
iris %>% mutate(sumVar = select(., contains("Sepal")) %>% rowSums()) %>% head
```

```
$index
```

```
# A tibble: 3 x 2
```

	var1	var2
	<chr>	<chr>
1	sex.factor	sex.factor2
2	age.factor	age.factor2
3	sex.factor2	age.factor2

```
$counts
```

```
$counts[[1]]
```

```
# A tibble: 2 x 3
```

	sex.factor	sex.factor2	n
	<fct>	<fct>	<int>
1	Female	M	445
2	Male	F	484

```
$counts[[2]]
```

```
# A tibble: 3 x 3
```

	age.factor	age.factor2	n
	<fct>	<fct>	<int>
1	<40 years	<60 years	70
2	40-59 years	<60 years	344
3	60+ years	60+ years	515

```
$counts[[3]]
```

```
# A tibble: 4 x 3
```

	sex.factor2	age.factor2	n
	<fct>	<fct>	<int>
1	M	<60 years	204
2	M	60+ years	241
3	F	<60 years	210

4 F                    60+ years            274

iRenameColumn.R

iSelectColumn.R

<= 22 Low

>= 23 & <= 41 Average

>=42 High

## 1 impute

### 1.1 impute continious

### 1.2 impute categorical

### 1.3 impute outlier

## 2 transform

### 2.1 min -max

### 2.2 skewness

### 2.3 log

## 3 binning

### 3.1 optimal binning

### 3.2 standardize

## 4 data transformation report

## 5 inspectdf

## 6 Descriptive Statistics

### 6.1 Table 1

---

### 6.2 Categorical Variables

#### 6.2.1 Split-Group Stats Categorical

#### 6.2.2 Grouped Categorical

---

## **6.3 Continuous Variables**

### **6.3.1 Split-Group Stats Continuous**

### **6.3.2 Grouped Continuous**

## 7 Cross Tables

## **8 Plots**

### **8.1 Categorical Variables**

## **9 Plots**

### **9.1 Continious Variables**

## 10 Survival Analysis

## 11 Pairwise comparison

## 12 Multivariate Analysis Survival

---

---

## 13 KM plot

```
explanatory = c("perfor.factor") dependent = "Surv(time, status)" colon_s %>% surv.plot(dependent, explanatory, xlab="Time (days)",  
pval=TRUE, legend="none")
```

Notes

Use `Hmisc::label()` to assign labels to variables for tables and plots.

```
label(colon_s$age.factor) = "Age (years)"
```

∞

Export dataframe tables directly or to R Markdown using `knitr::kable()`.

Note wrapper `summary.missing()` can be useful. Wraps `mice::md.pattern`.

```
colon_s %>% summary.missing(dependent, explanatory)
```



Where a multivariable model contains a subset of the variables specified in the full univariable set, this can be specified.

```
explanatory = c("age.factor", "sex.factor", "obstruct.factor", "perfor.factor") explanatory.multi = c("age.factor", "obstruct.factor") dependent = 'mort_5yr' colon_s %>% summarizer(dependent, explanatory, explanatory.multi)
```

Random effects.

```
e.g. lme4::glmer(dependent ~ explanatory + (1 | random_effect), family="binomial")
```

```
explanatory = c("age.factor", "sex.factor", "obstruct.factor", "perfor.factor") explanatory.multi = c("age.factor", "obstruct.factor") random.effect = "hospital" dependent = 'mort_5yr' colon_s %>% summarizer(dependent, explanatory, explanatory.multi, random.effect)
```

metrics=TRUE provides common model metrics.

```
colon_s %>% summarizer(dependent, explanatory, explanatory.multi, metrics=TRUE)
```

Cox proportional hazards

```
e.g. survival::coxph(dependent ~ explanatory)
```

```
explanatory = c("age.factor", "sex.factor", "obstruct.factor", "perfor.factor") dependent = "Surv(time, status)"
```

```
colon_s %>% summarizer(dependent, explanatory)
```

Rather than going all-in-one, any number of subset models can be manually added on to a summary.factorlist() table using summarizer.merge(). This is particularly useful when models take a long-time to run or are complicated.

Note requirement for glm.id=TRUE. fit2df is a subfunction extracting most common models to a dataframe.

```
explanatory = c("age.factor", "sex.factor", "obstruct.factor", "perfor.factor") explanatory.multi = c("age.factor", "obstruct.factor") random.effect = "hospital" dependent = 'mort_5yr'
```

## 14 Separate tables

```
colon_s %>% summary.factorlist(dependent, explanatory, glm.id=TRUE) -> example.summary
```

```
colon_s %>% glmuni(dependent, explanatory) %>% fit2df(estimate.suffix=" (univariable)") -> example.univariable
```

```
colon_s %>% glmmulti(dependent, explanatory) %>% fit2df(estimate.suffix=" (multivariable)") -> example.multivariable
```

```
colon_s %>% glmmixed(dependent, explanatory, random.effect) %>% fit2df(estimate.suffix=" (multilevel)") -> example.multilevel
```

## 15 Pipe together

```
example.summary %>% summarizer.merge(example.univariable) %>% summarizer.merge(example.multivariable) %>% summarizer.merge(example.multilevel) %>% select(-c(glm.id, index)) -> example.final
```

Cox Proportional Hazards example with separate tables merged together.

```
explanatory = c("age.factor", "sex.factor", "obstruct.factor", "perfor.factor") explanatory.multi = c("age.factor", "obstruct.factor") dependent = "Surv(time, status)"
```

## 16 Separate tables

```
colon_s %>% summary.factorlist(dependent, explanatory, glm.id=TRUE) -> example2.summary  
colon_s %>% coxphuni(dependent, explanatory) %>% fit2df(estimate.suffix=" (univariable)") -> exam-  
ple2.univariable  
colon_s %>% coxphmulti(dependent, explanatory.multi) %>% fit2df(estimate.suffix=" (multivariable)") ->  
example2.multivariable
```

## 17 Pipe together

```
example2.summary %>% summarizer.merge(example2.univariable) %>% summarizer.merge(example2.mul-  
tivariable) %>% select(-c(glm.id, index)) -> example2.final example2.final
```

## 18 OR plot

```
explanatory = c("age.factor", "sex.factor", "obstruct.factor", "perfor.factor") dependent = 'mort_5yr'  
colon_s %>% or.plot(dependent, explanatory) # Previously fitted models (glmmulti() or glmmixed())  
can be provided directly to glmfit
```

## 19 HR plot (not fully tested)

```
explanatory = c("age.factor", "sex.factor", "obstruct.factor", "perfor.factor") dependent = "Surv(time, sta-  
tus)" colon_s %>% hr.plot(dependent, explanatory, dependent_label = "Survival") # Previously fitted  
models (coxphmulti) can be provided directly using coxfit
```

## 20 ANOVA

## 21 Save Final Data

## 22 Final Data Summary

## 23 Software and Libraries Used

The jamovi project (2019). jamovi. (Version 0.9) [Computer Software]. Retrieved from <https://www.jamovi.org>. R Core Team (2018). R: A Language and environment for statistical computing. [Computer software]. Retrieved from <https://cran.r-project.org/>. Fox, J., & Weisberg, S. (2018). car: Companion to Applied Regression. [R package]. Retrieved from <https://cran.r-project.org/package=car>.

## 24 Session Info

## 25 Notes

Last update on 2019-11-06 12:29:44

Serdar Balci, MD, Pathologist

[drserdarbalci@gmail.com](mailto:drserdarbalci@gmail.com)

<https://rpubs.com/sbalci/CV>