Extinction Law

Let's take a look at the extinction laws that we can use to de-redden the young stars in Cep Ob3b. The three main laws are the Rieke and Lebofsky (1985), the Cardelli, Clayton, and Mathis (1989), and Allen et al. (2014) link. The spectra used in Allen et al. (2014) were recently calssified visually and many that were used as K5 background giants have been retyped. We will explore here how reasonable the (**???**) law is.

We will make use of the reticulate package. Note, we are defining the python engine in the R setup chunk.

```
#devtools::install_github("rstudio/reticulate")
library(reticulate)
use_python("/anaconda3/bin/python")
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

We read in the data set. Here we will start with the `full.df.csv` dataset. For details about this data set see the data documentation. We are interested in the spectral typing of the stars with *Hectospec* spectra.

```
data_path <- "/Users/thomasallen/cep_ob3b/cepr/data/"
data_path2 <- "/Users/thomasallen/cep_ob3b/data/"

#full.df.csv

full.df <- read_csv(paste(data_path2,"full.df.csv",sep=""))
```

```
## Warning: Missing column names filled in: 'X1' [1]

## Parsed with column specification:
## cols(
##   .default = col_double(),
##   X1 = col_integer(),
##   bmag = col_character(),
##   berr = col_character(),
##   vmag = col_character(),
##   verr = col_character(),
##   imag = col_character(),
##   ierr = col_character(),
##   cluster = col_character(),
##   cloud = col_character(),
##   disk = col_character(),
##   xray = col_character(),
##   acis = col_character(),
##   spec = col_character(),
```

```
##   chelle = col_character(),
##   spt = col_character(),
##   spterr = col_character(),
##   tio = col_character(),
##   tior = col_character(),
##   cah = col_character(),
##   cahr = col_character()
##   # ... with 28 more columns
## )
```

```
## See spec(...) for full column specifications.
```

```r
#head(full.df)
gsdss.df <- read_csv(paste(data_path,"gsdss.csv",sep=""))
```

```
## Parsed with column specification:
## cols(
##   mag = col_double(),
##   err = col_double()
## )
```

```r
rsdss.df <- read_csv(paste(data_path,"rsdss.csv",sep=""))
```

```
## Parsed with column specification:
## cols(
##   mag = col_double(),
##   err = col_double()
## )
```

```r
full.df <- full.df %>%
    mutate(gmag=gsdss.df$mag,gerr=gsdss.df$err) %>%
    mutate(rmag=rsdss.df$mag,rerr=rsdss.df$err)

full.df <- full.df %>%
    mutate(bmag=as.numeric(bmag)) %>%
    mutate(berr=as.numeric(berr)) %>%
    mutate(vmag=as.numeric(vmag)) %>%
    mutate(verr=as.numeric(verr)) %>%
    mutate(imag=as.numeric(imag)) %>%
    mutate(ierr=as.numeric(ierr))
```

We want the objects that have spectral types. These will be rows where the columns `spt`, the spectral type as classified by eye, and `spt_old`, the spectral type as classified by regression.

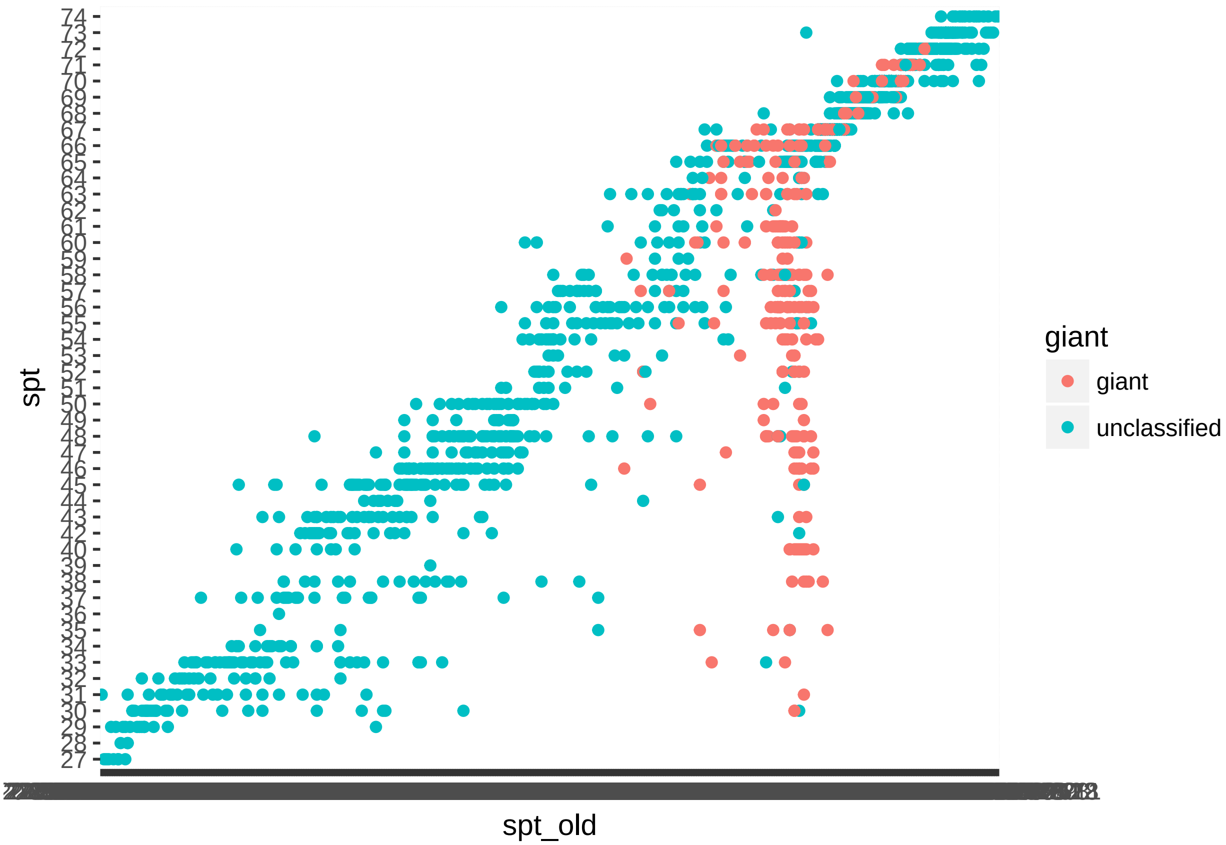
Lets make a column that tells us which stars we classified as probable background giants.

```r
full.df <- full.df %>%
    mutate(giant = ifelse(cagiant == "giant" | nagiant == "giant", "giant", "unclassified"))


spt.df <- full.df %>%
    filter(is.na(spt)==FALSE & is.na(spt_old)==FALSE)


head(spt.df)
```

```
## # A tibble: 6 x 93
##      X1    ra   dec  bmag  berr  vmag  verr  imag  ierr  jmag   jerr  hmag
```

```
##   <int> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl> <dbl>
## 1  4666  343.  62.4    NA    NA    NA    NA    NA    NA  13.8 0.025   12.4
## 2  4760  344.  62.3    NA    NA    NA    NA    NA    NA  13.8 0.025   13.1
## 3  4925  343.  62.4    NA    NA    NA    NA    NA    NA  14.8 0.046   13.7
## 4  5949  343.  62.4    NA    NA    NA    NA    NA    NA  15.3 0.0580  14.2
## 5  6017  344.  62.4    NA    NA    NA    NA    NA    NA  13.6 0.035   13.0
## 6  7049  344.  62.4    NA    NA    NA    NA    NA    NA  13.0 0.023   12.6
## # ... with 81 more variables: herr <dbl>, kmag <dbl>, kerr <dbl>,
## #   c1mag <dbl>, c1err <dbl>, c2mag <dbl>, c2err <dbl>, c3mag <dbl>,
## #   c3err <dbl>, c4mag <dbl>, c4err <dbl>, m24mag <dbl>, m24err <dbl>,
## #   cluster <chr>, cloud <chr>, disk <chr>, xray <chr>, acis <chr>,
## #   spec <chr>, chelle <chr>, spt <chr>, spterr <chr>, tio <chr>,
## #   tior <chr>, cah <chr>, cahr <chr>, spt_old <chr>, spterr_old <chr>,
## #   nagiant <chr>, cagiant <chr>, minxray.ra <int>, minxray.dec <int>,
## #   minxray.id <chr>, minxray.rcnts <int>, minxray.ncnts <int>,
## #   minxray.npflux <int>, minxray.npfluxerr <int>, minxray.nh <int>,
## #   minxray.nherr <int>, minxray.kt1 <int>, minxray.kt1err <int>,
## #   minxray.aflux <int>, minxray.uflux <int>, minxray.rchi <int>,
## #   medxray.ra <dbl>, medxray.dec <dbl>, medxray.id <chr>,
## #   medxray.rcnts <int>, medxray.ncnts <dbl>, medxray.npflux <dbl>,
## #   medxray.npfluxerr <dbl>, medxray.nh <dbl>, medxray.nherr <dbl>,
## #   medxray.kt1 <int>, medxray.kt1err <int>, medxray.aflux <dbl>,
## #   medxray.uflux <dbl>, medxray.rchi <dbl>, maxxray.ra <dbl>,
## #   maxxray.dec <dbl>, maxxray.id <chr>, maxxray.rcnts <int>,
## #   maxxray.ncnts <dbl>, maxxray.npflux <dbl>, maxxray.npfluxerr <dbl>,
## #   maxxray.nh <dbl>, maxxray.nherr <dbl>, maxxray.kt1 <dbl>,
## #   maxxray.kt1err <dbl>, maxxray.aflux <dbl>, maxxray.uflux <dbl>,
## #   maxxray.rchi <dbl>, lbol.teff.sa.lbol <chr>, lbol.teff.sa.teff <chr>,
## #   lbol.teff.sa.sa <chr>, member <chr>, gmag <dbl>, gerr <dbl>,
## #   rmag <dbl>, rerr <dbl>, giant <chr>
```

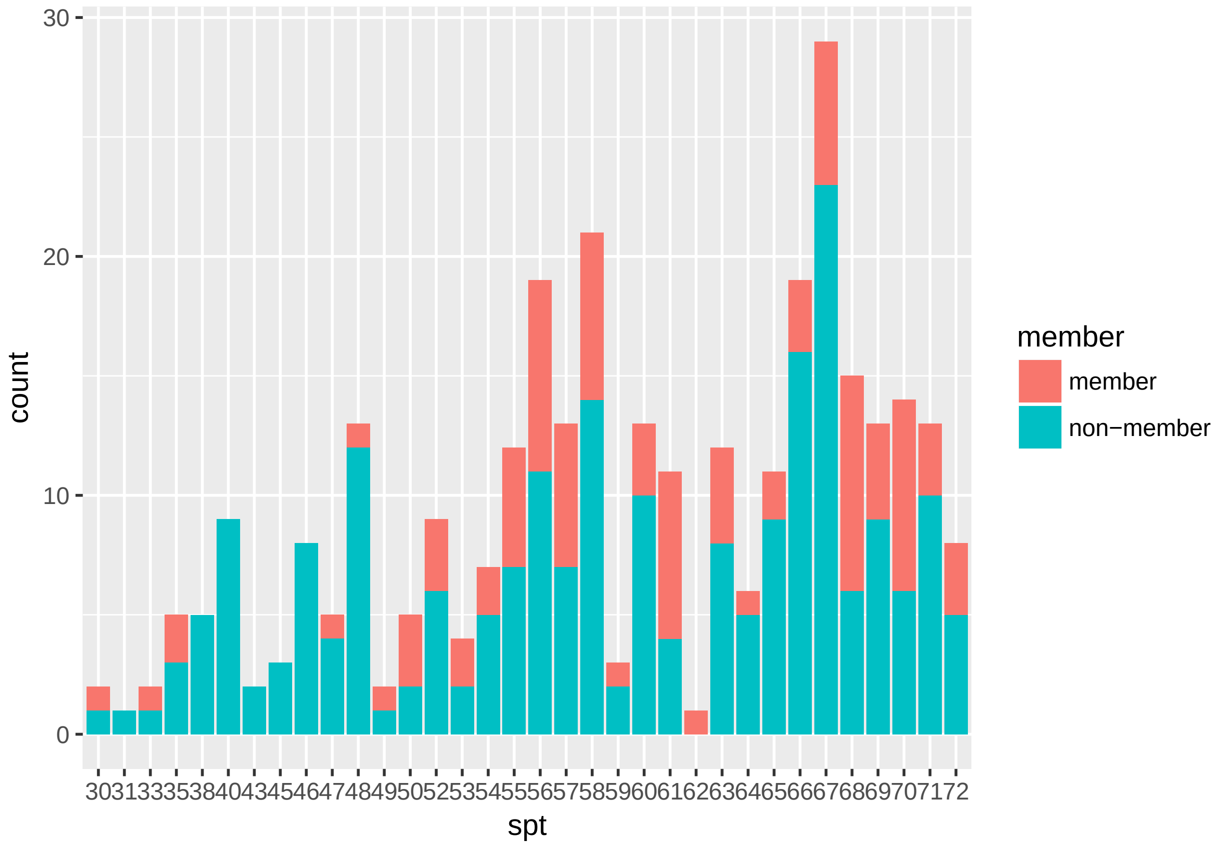
```r
plot1 <- spt.df %>% ggplot(aes(x=spt_old,y=spt,color=giant)) +
    geom_point()


plot1
```



It looks like many of the stars classified as K5/K6 background giants have the wrong spectral classification.

Lets look at the distibution of visually determined spectral types of these “background giants”.

```
spt.df %>%
    filter(giant=="giant") %>%
    ggplot(aes(x=spt,fill=member)) + geom_bar()
```



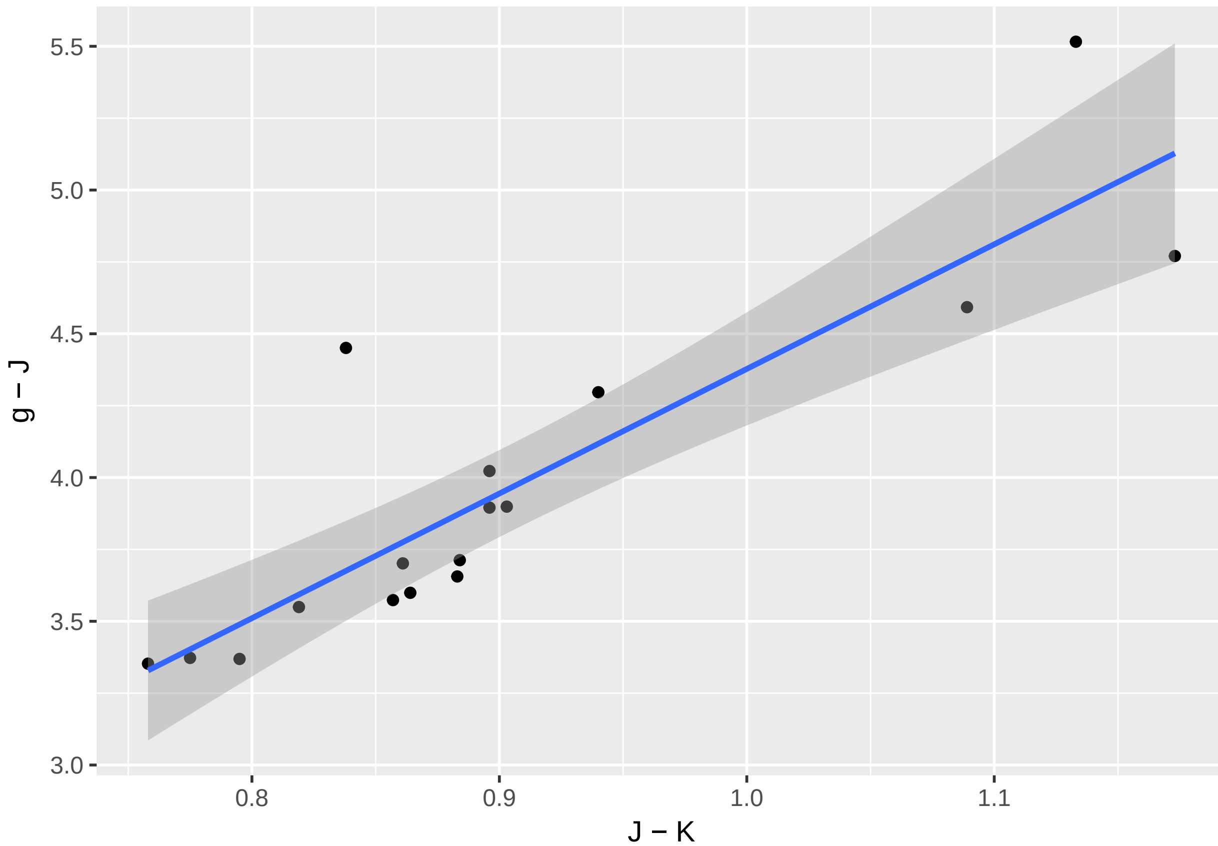
Ok, so after removing potential members, there are about 25 K5/K6 stars that still remain, and over 20 K7s.

```
ext_cc_plot_fit <- function(df,c1=vmag,c2=jmag,c3=kmag,xtitle="J - K",ytitle="V - J")  {

    # Plots X - J vs. J - K and fits a linear model
    # Slope of linear model is used to derive Ax/Aj extinction coefficient

    c1<-enquo(c1)
    c2<-enquo(c2)
    c3<-enquo(c3)

    df.plot <- df %>%
        filter(giant=="giant") %>%
        filter(member=="non-member") %>%
        filter(is.na(!!c2)==FALSE & is.na(!!c3)==FALSE & is.na(!!c1)==FALSE) %>%
        filter(spt==65 | spt==66) %>% # | spt==67)


        #mutate(c1=as.numeric(c1)) %>%
        mutate(x=!!c2 - !!c3,y= !!c1 - !!c2) %>%
        select(x,y)
    #filter(jmk < 1.75)


    plot <- df.plot %>%
        ggplot(aes(x=x,y=y)) +
        geom_point() +
```

```r
        geom_smooth(method='lm') +
        labs(x=xtitle,y=ytitle)

    fit <- lm(y ~ x, data = df.plot)
    print(summary(fit))
    #print(plot)

    return(plot)
  #
}
```

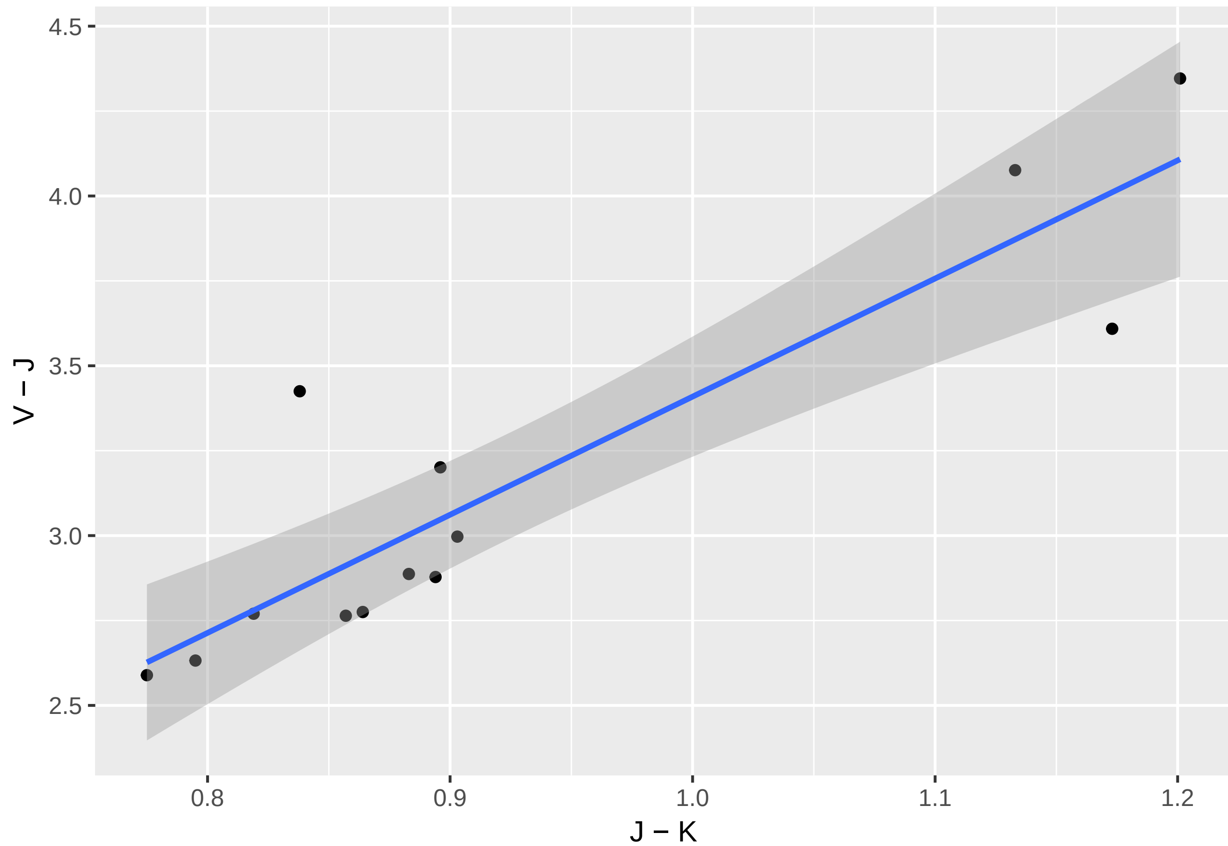
```r
# g - J

spt.df %>% ext_cc_plot_fit(c1=gmag,ytitle="g - J")
```

```
##
## Call:
## lm(formula = y ~ x, data = df.plot)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.35719 -0.17104 -0.05845  0.02387  0.77533
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.04265    0.55792   0.076     0.94
## x            4.33510    0.61231   7.080 3.75e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2926 on 15 degrees of freedom
## Multiple R-squared:  0.7697, Adjusted R-squared:  0.7543
## F-statistic: 50.12 on 1 and 15 DF,  p-value: 3.747e-06
```



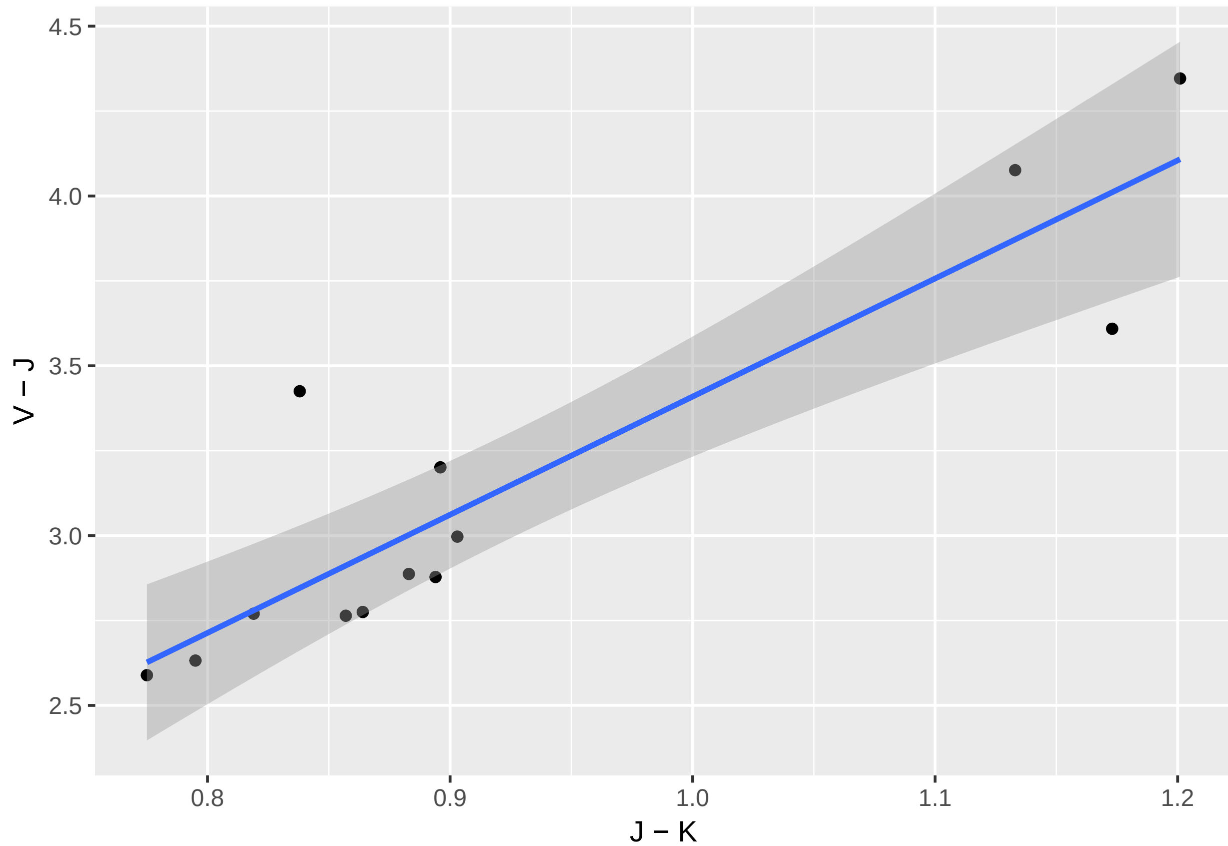

```
# V - J

spt.df %>% ext_cc_plot_fit(c1=vmag,ytitle="V - J")
```

```
##
## Call:
## lm(formula = y ~ x, data = df.plot)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.40187 -0.14781 -0.06417  0.15355  0.57927
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.06887    0.47660  -0.144    0.888
## x            3.47804    0.50926   6.830 2.84e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2554 on 11 degrees of freedom
## Multiple R-squared:  0.8092, Adjusted R-squared:  0.7918
## F-statistic: 46.64 on 1 and 11 DF,  p-value: 2.84e-05
```



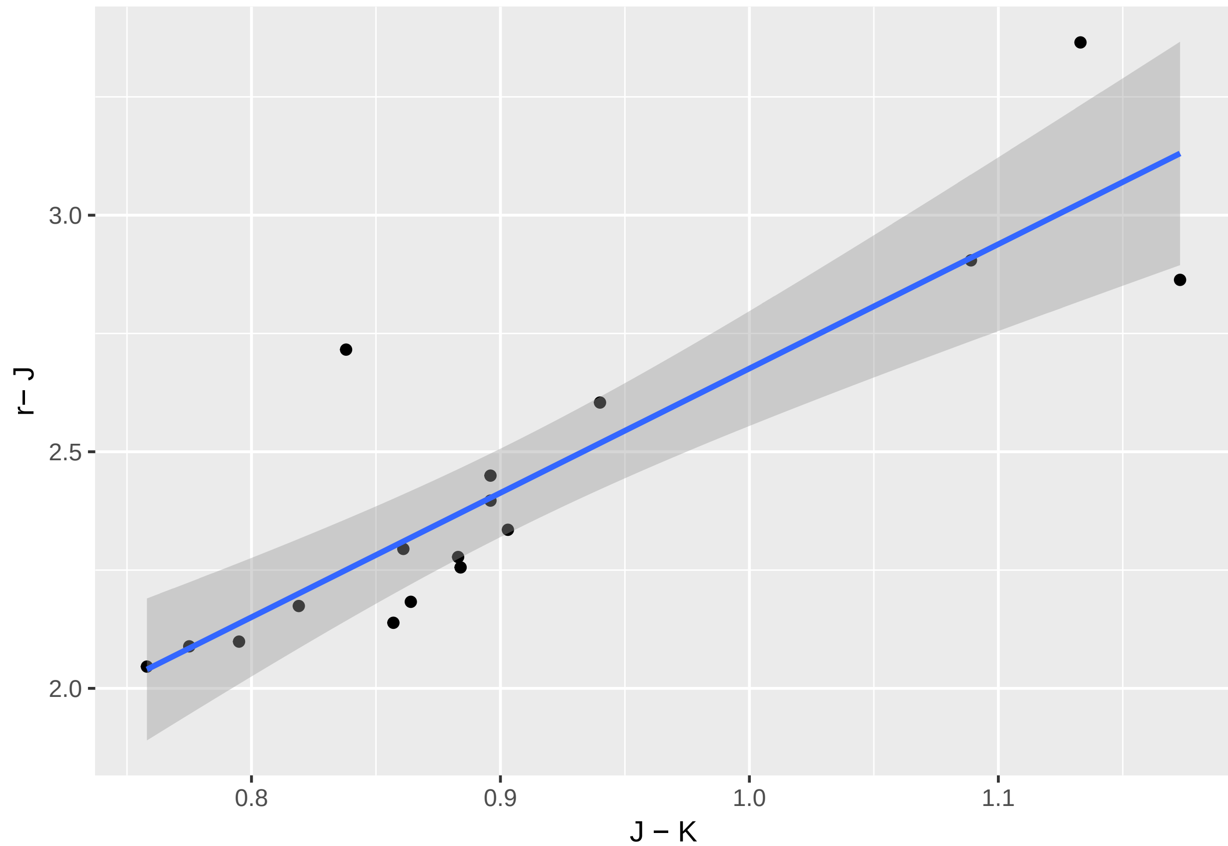

```r
# V - J
# Filter outlier

spt.df %>%
    filter(jmag - kmag < 1.75) %>%
    ext_cc_plot_fit(c1=vmag,ytitle="V - J")
```

```
##
## Call:
## lm(formula = y ~ x, data = df.plot)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.40187 -0.14781 -0.06417  0.15355  0.57927
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.06887    0.47660  -0.144    0.888
## x            3.47804    0.50926   6.830 2.84e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2554 on 11 degrees of freedom
## Multiple R-squared:  0.8092, Adjusted R-squared:  0.7918
## F-statistic: 46.64 on 1 and 11 DF,  p-value: 2.84e-05
```




```
# r - J

spt.df %>% ext_cc_plot_fit(c1=rmag,ytitle="r- J")
```

```
##
## Call:
## lm(formula = y ~ x, data = df.plot)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.26718 -0.09085 -0.01599  0.00600  0.46569
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.04814    0.34434   0.140    0.891
## x            2.62776    0.37791   6.953 4.63e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1806 on 15 degrees of freedom
## Multiple R-squared:  0.7632, Adjusted R-squared:  0.7474
## F-statistic: 48.35 on 1 and 15 DF,  p-value: 4.627e-06
```




```r
alam <- function(slope) {

    alam <- slope * (1 - 0.397) + 1

    return(alam)

}

alam(4.3)
```

```
## [1] 3.5929
```

```r
alam(3.5)
```

```
## [1] 3.1105
```

```r
alam(2.6)
```

```
## [1] 2.5678
```


Allen, T. S., J. J. Prchlik, S. T. Megeath, R. A. Gutermuth, J. L. Pipher, T. Naylor, and R. D. Jeffries. 2014. "An Anomalous Extinction Law in the Cep OB3b Young Cluster: Evidence for Dust Processing During Gas Dispersal." *Apj* 786 (May): 113. doi:10.1088/0004-637X/786/2/113.

Cardelli, J. A., G. C. Clayton, and J. S. Mathis. 1989. "The Relationship Between Infrared, Optical, and Ultraviolet Extinction." *Apj* 345 (October): 245–56. doi:10.1086/167900.

Rieke, G. H., and M. J. Lebofsky. 1985. "The Interstellar Extinction Law from 1 to 13 Microns." *Apj* 288 (January): 618–21. doi:10.1086/162827.