Supporting information for:

Dynamic Proteoform-Resolved Profiling of Plasminogen Activation Reveals Novel Primary N-Terminal Cleavage Site

Dario A. T. Cramer[1,2], Victor Yin[1,2], Tomislav Caval[1,2], Vojtech Franc[1,2], Dingyi Yu[3], Guojie Wu[4], Gordon Lloyd[4], Christopher Langendorf[3], James C. Whisstock[4], Ruby H. P. Law[4 #], Albert J. R. Heck[1,2 #]

[1]Biomolecular Mass Spectrometry and Proteomics, Bijvoet Center for Biomolecular Research and Utrecht Institute for Pharmaceutical Science, University of Utrecht, Padualaan 8, Utrecht, 3584 CH, The Netherlands

[2]Netherlands Proteomics Centre, University of Utrecht, Padualaan 8, Utrecht, 3584 CH, The Netherlands

[3]St Vincent's Institute of Medical Research, Victoria, 3065 Australia

[4]Department of Biochemistry and Molecular Biology, Monash University, Clayton, Melbourne, VIC 3800 Australia

[#]Corresponding Author: a.j.r.heck@uu.nl

This supporting information contains:

Supplementary tables

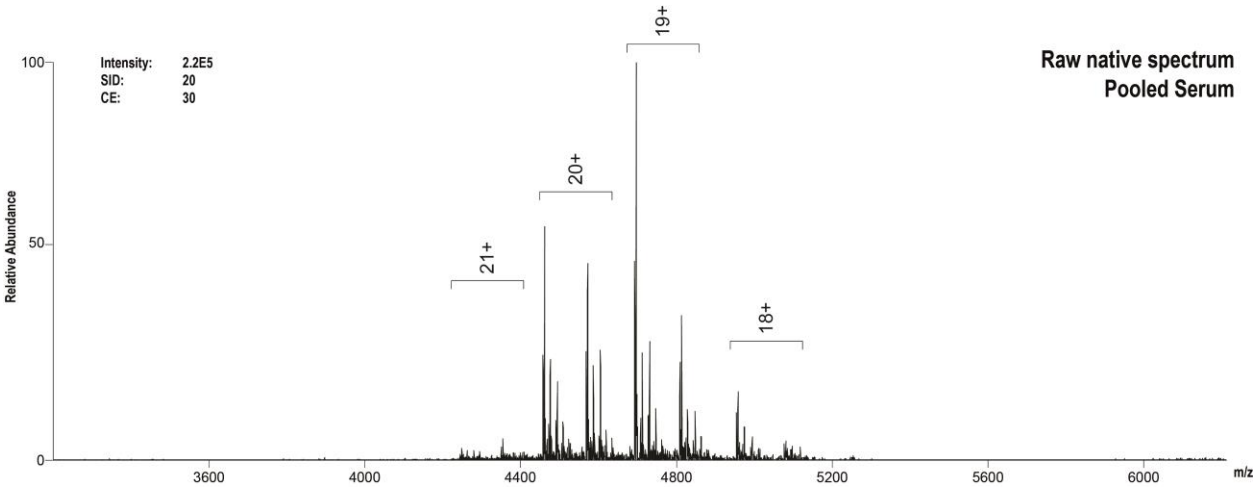
Suppl. table 1: annotation of all observed proteoforms

Suppl. table 2: masses of peptides found during Plg activation

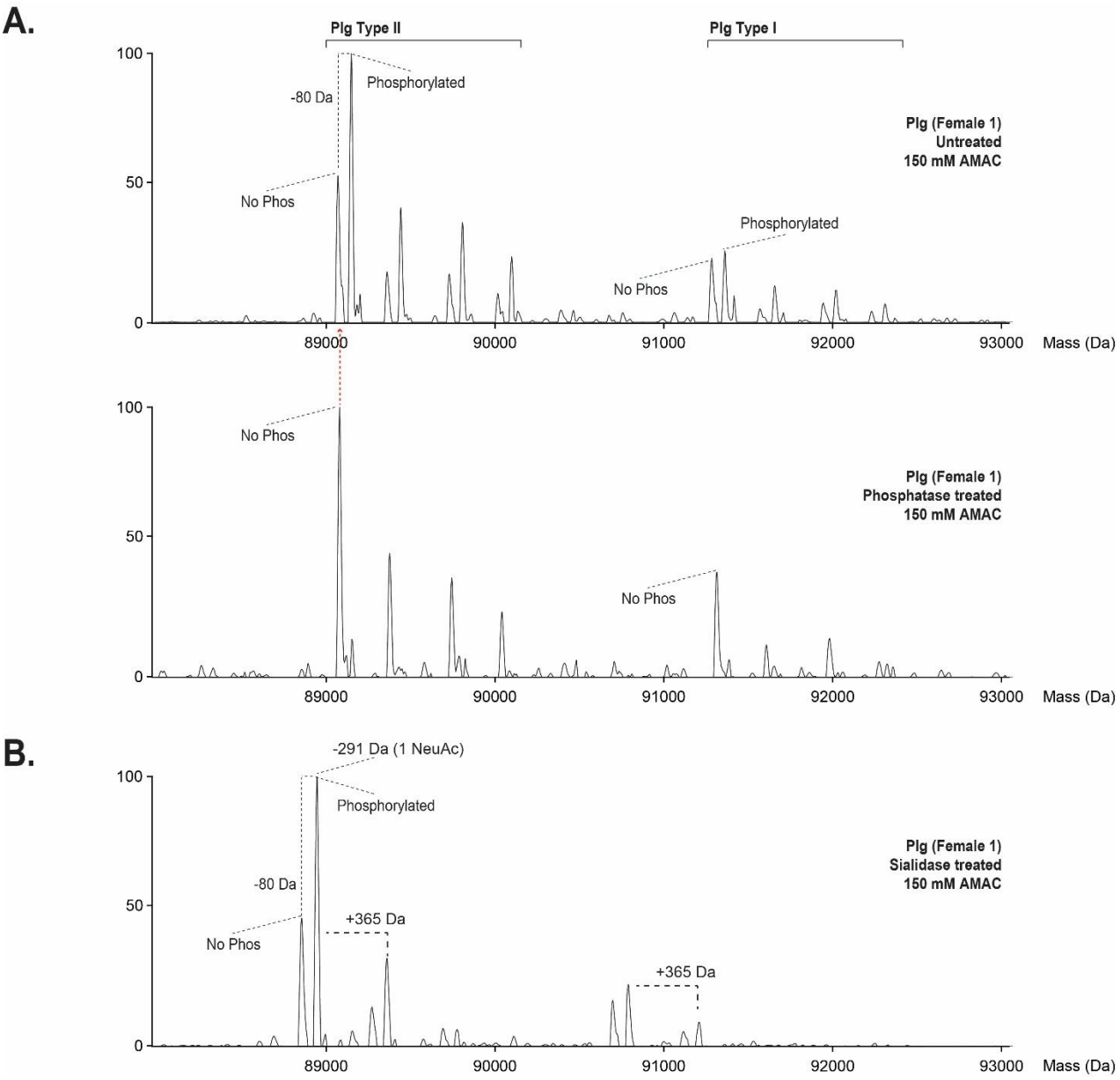
Suppl. table 3: peptide identification for annotation of Plg

Suppl. table 4: quantification of PTMs using peptide-centric MS data

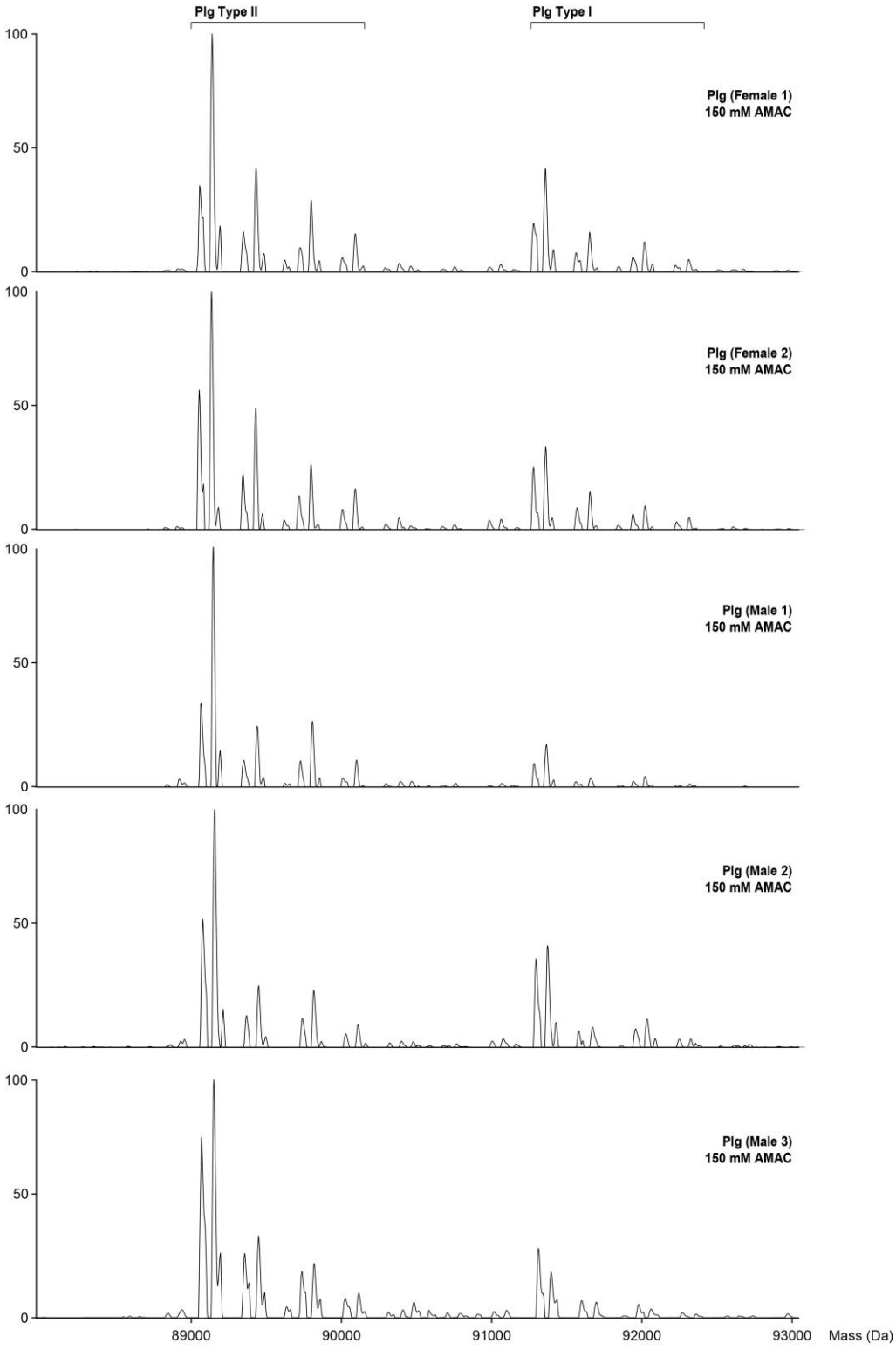
Suppl. table 5: selected genes for phylogenetic analysis



Supplementary figure 1: native MS spectrum of Plg reveals a charge state range of +18 to +20 with two recognizable series of proteoforms per charge state.

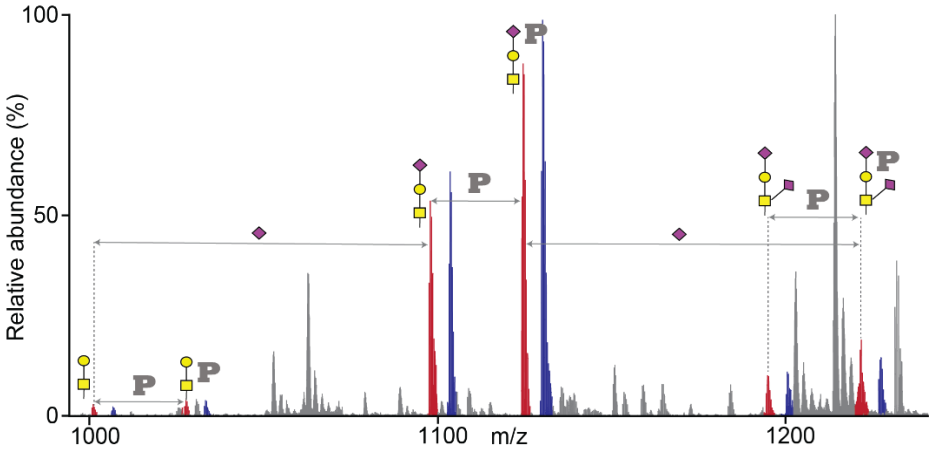


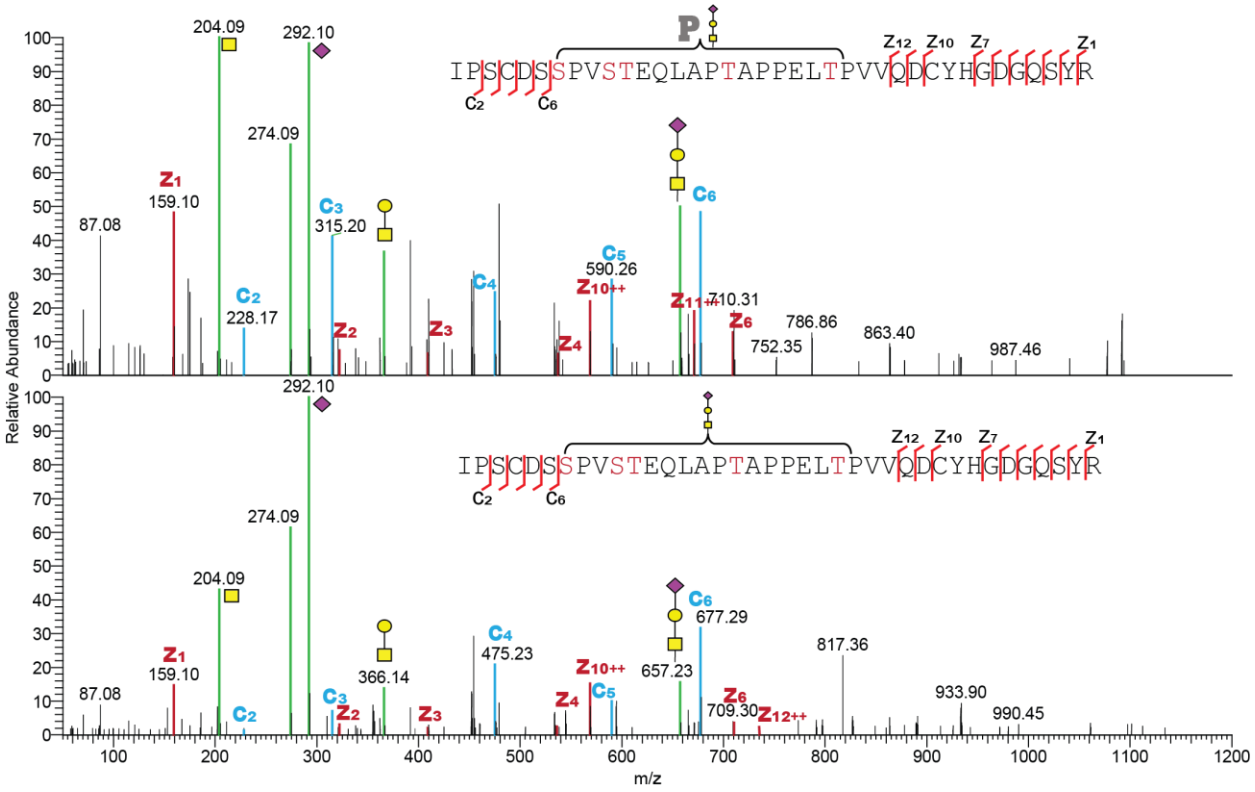
Supplementary figure 2: A. treatment with phosphatase collapses all annotated peak pairs into single proteoforms. The mass loss of ~80 Da supports the annotation that all glycoforms occurred as partially phosphorylated. **B**. treatment with sialidase removes up to two NeuAc on type II Plg and up to four glycans on type I Plg. The removal of two additional NeuAc on type I Plg supports the annotation of a complex, biantennary N-glycan. Phosphorylation of Plg is unaffected by treatment with sialidase alone.



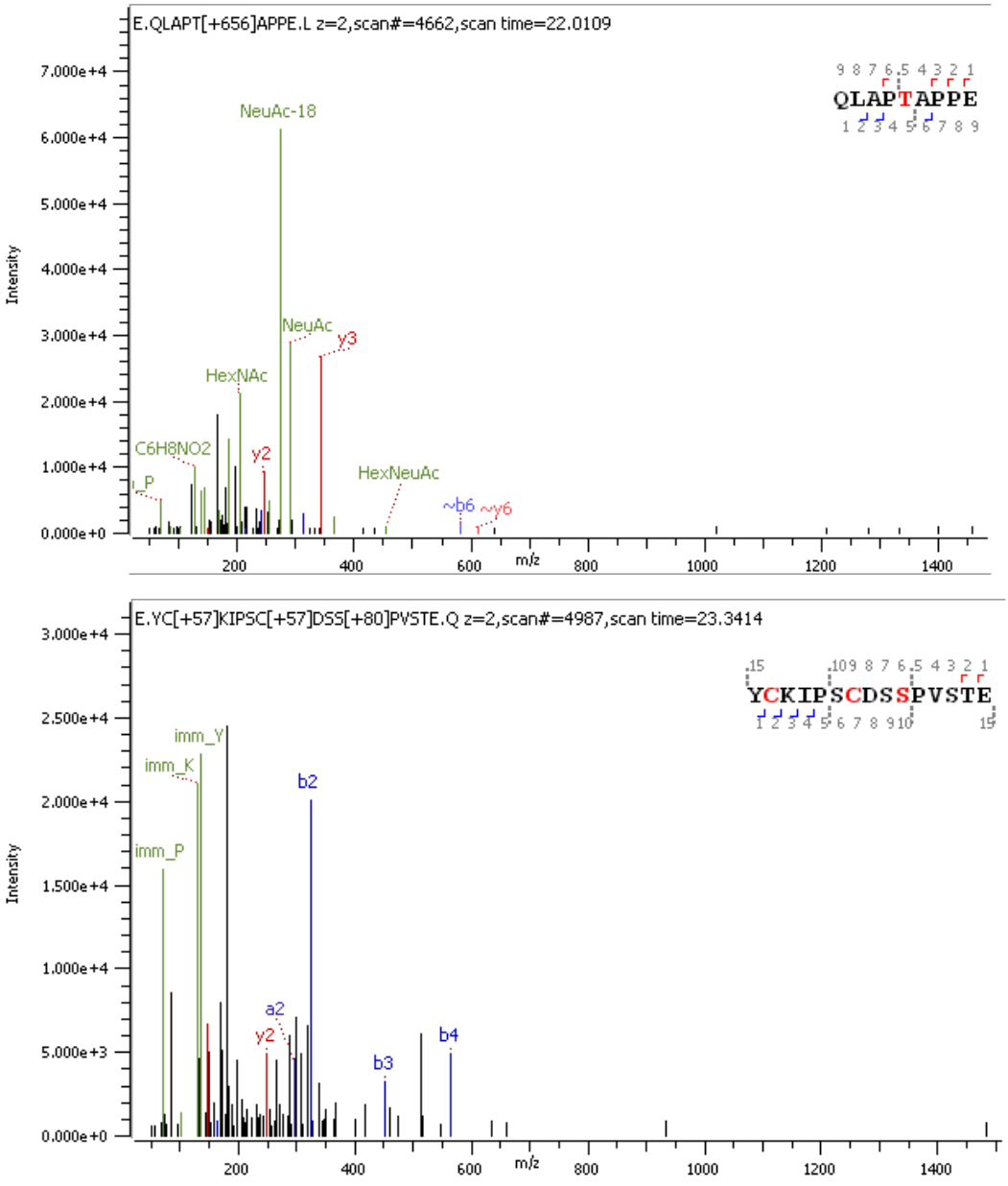
Supplementary figure 3: a comparison of native and deconvoluted spectra of five individual donors (2 female, 3 male) reveals that phosphorylation and glycosylation are a consistent feature of Plg.

A.

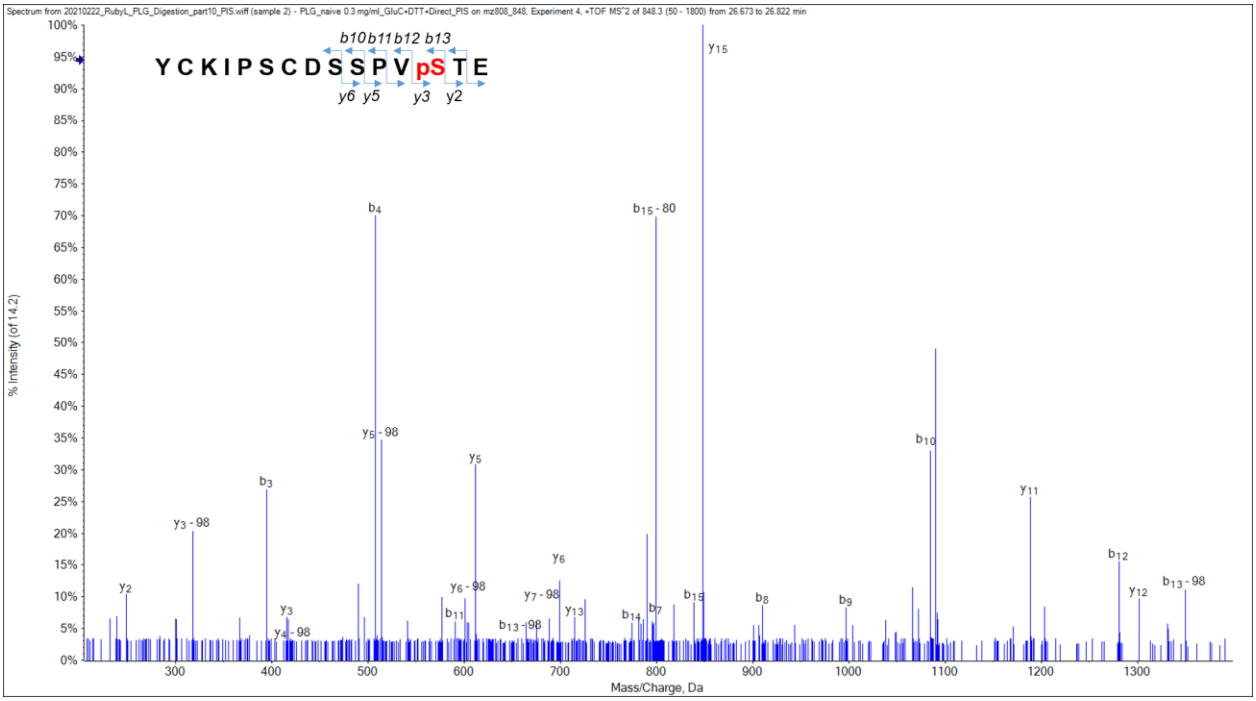

Relative abundance (%)
100
50
0
1000
1100
m/z
1200
P


Relative Abundance
m/z
IPSCDSSPVSTEQLAPTAPPELTPVVQDCYHGDGQSYR
C2
C6
Z12
Z10
Z7
Z1
204.09
292.10
274.09
87.08
159.10
315.20
228.17
590.26
710.31
786.86
752.35
863.40
987.46
366.14
475.23
657.23
677.29
817.36
709.30
933.90
990.45

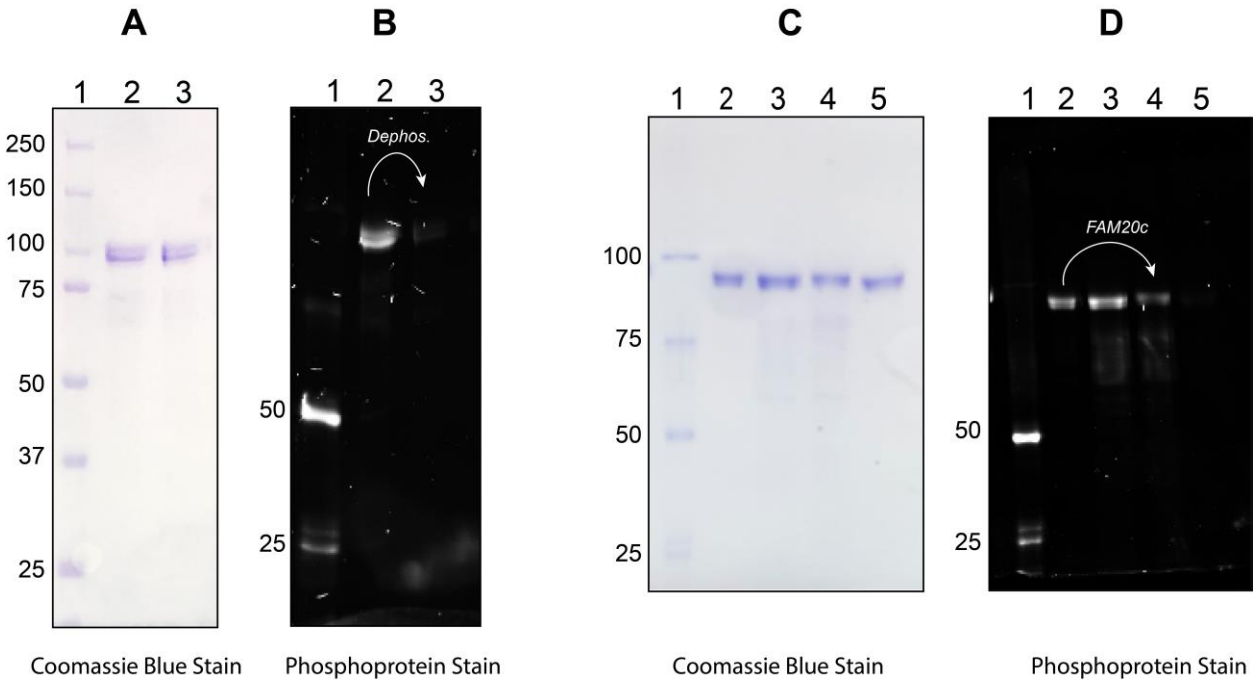
B.


E.QLAPT[+656]APPE.L z=2,scan#=4662,scan time=22.0109
QLAPTAPPE
NeuAc-18
NeuAc
y3
HexNAc
C6H8NO2
y2
HexNeuAc
~b6
~y6
Intensity
m/z
E.YC[+57]KIPSC[+57]DSS[+80]PVSTE.Q z=2,scan#=4987,scan time=23.3414
YCKIPSCDSSPVSTE
imm_Y
imm_K
b2
imm_P
a2
y2
b4
b3
Intensity
m/z

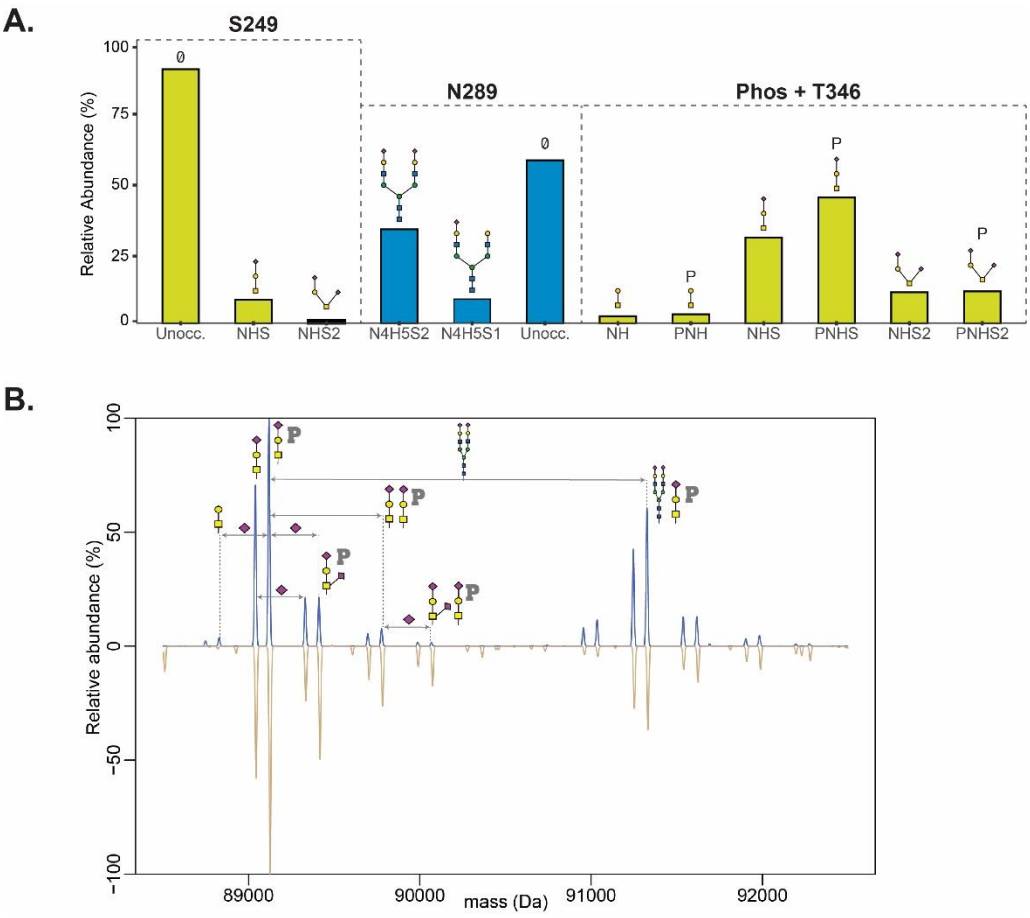
C.



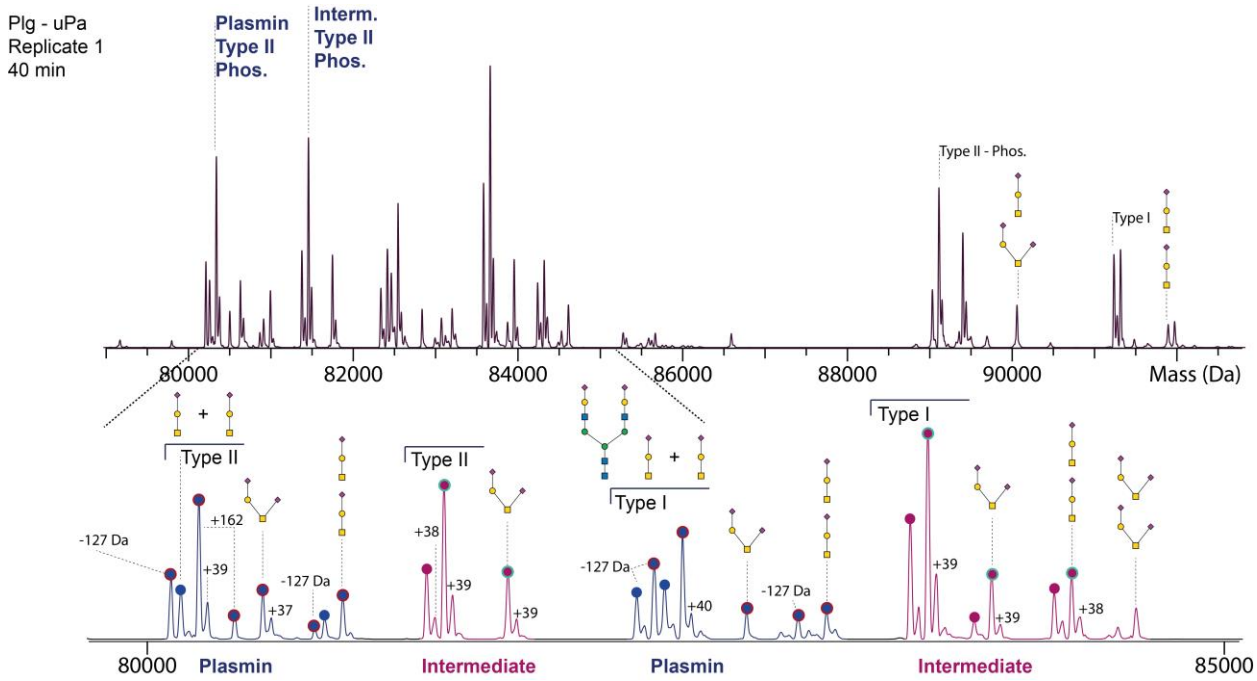
Supplementary figure 4: bottom-up MS2 spectra on an Orbitrap mass analyzer show **A**. ms1 identification of a phosphorylated peptide containing the thr346 o-glycan with no ms2 coverage. **B**. glu-c peptides of O-glycosylation and phosphorylation on the same peptide and **C**. direct fragmentation of the phosphorylated serine (Ser339) on a TOF mass analyzer.



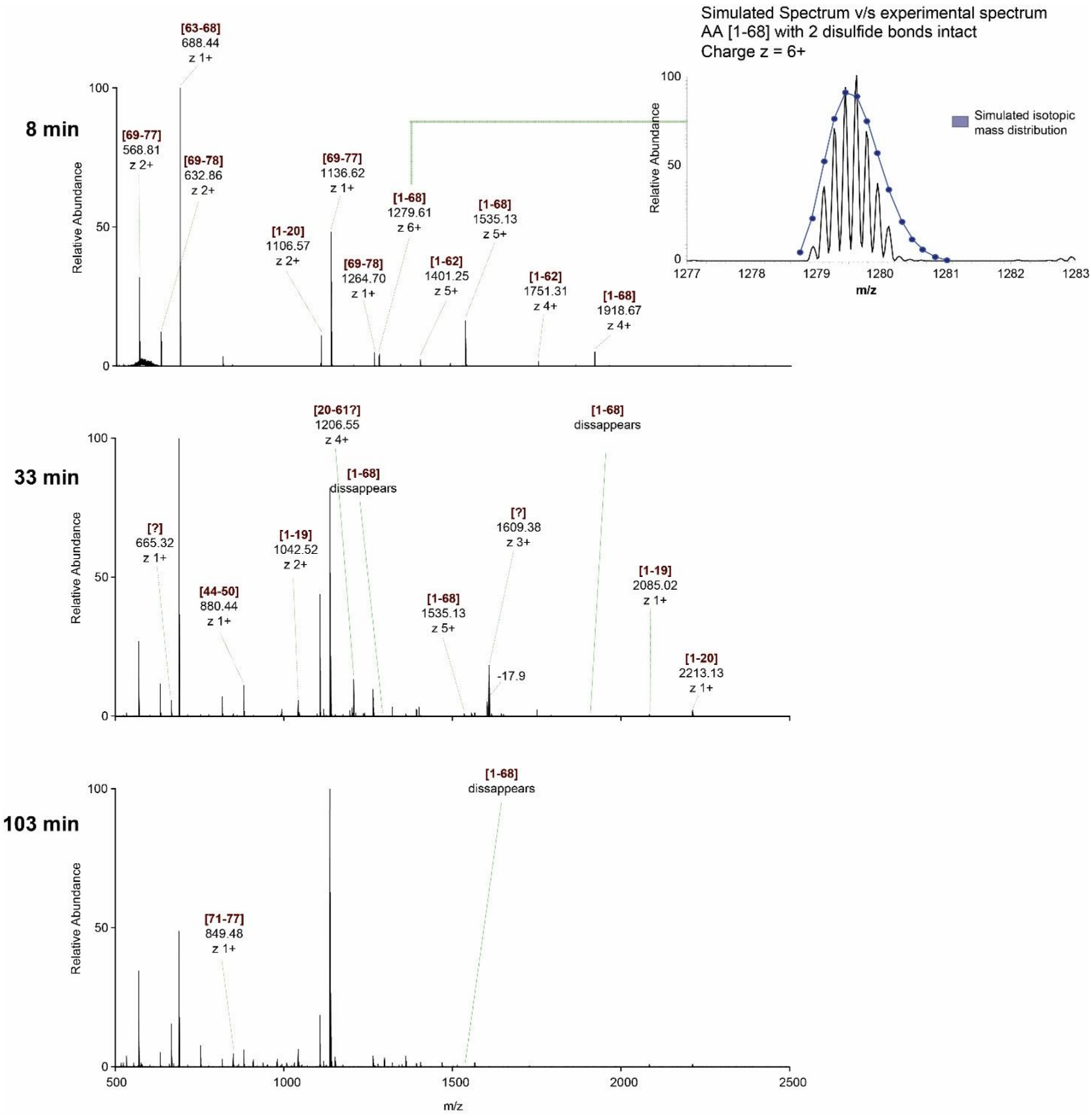
Supplementary figure 5: A. Coomassie stain and **B**. phosphoprotein stain of native Plg (Lanes 2), which is dephosphorylated with Lambda protein phosphatase (Lanes 3). **C.** Coomassie stain and **D**. phosphoprotein stain of native Plg (Lanes 2), which is treated with Fams20C before (Lanes 3) or after (Lanes 4) dephosphorylation (Lanes 5). Also shown is the molecular weight maker (Lanes 1).



Supplementary figure 6: A. Relative abundances of peptides carrying all Plg PTMs. The O-glycan site Thr346 is fully occupied, the O-glycan site Ser249 is rarely occupied, the N-glycan site Asn289 is partially occupied and Plg is partially phosphorylated, likely at Ser339. **B**. Experimental (top) and simulated (bottom) native MS spectrum, whereby the latter is based on the quantitative bottom-up MS data. The resemblance of both data sets demonstrates their correctness and completeness. The quantification data can be found in suppl. table 4.

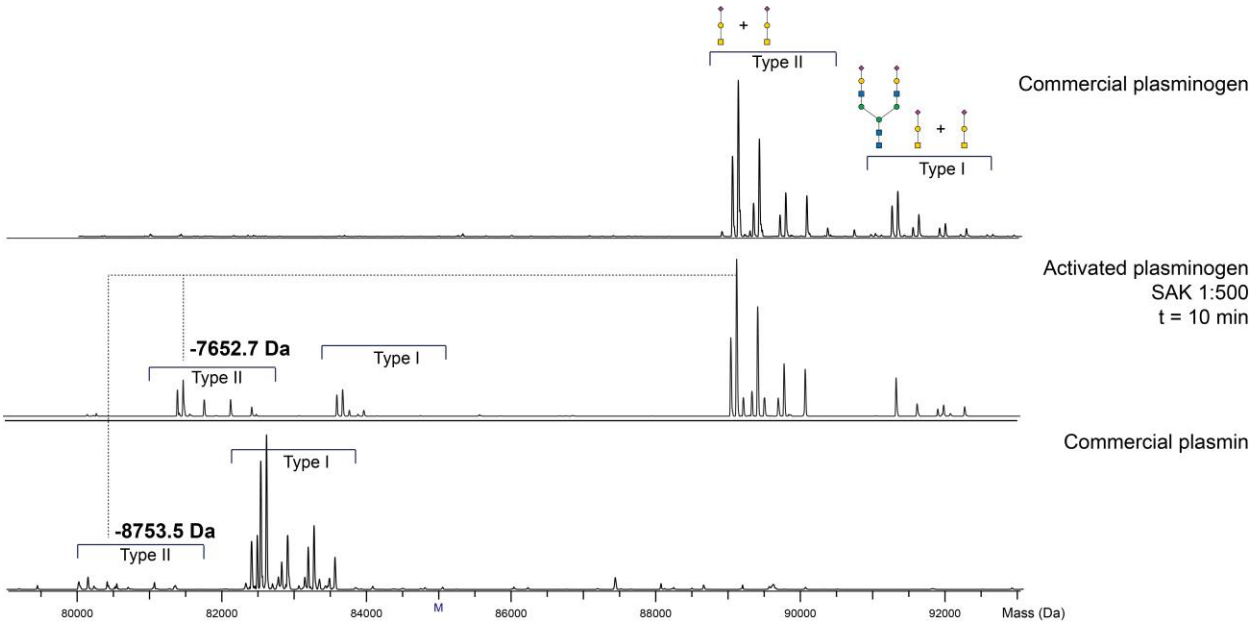


Supplementary figure 7: raw native spectrum and deconvoluted spectrum of partially activated Plg shows the resolution of individual proteoforms. Three species, all with a highly similar proteoform profile, are detectable without prior separation.

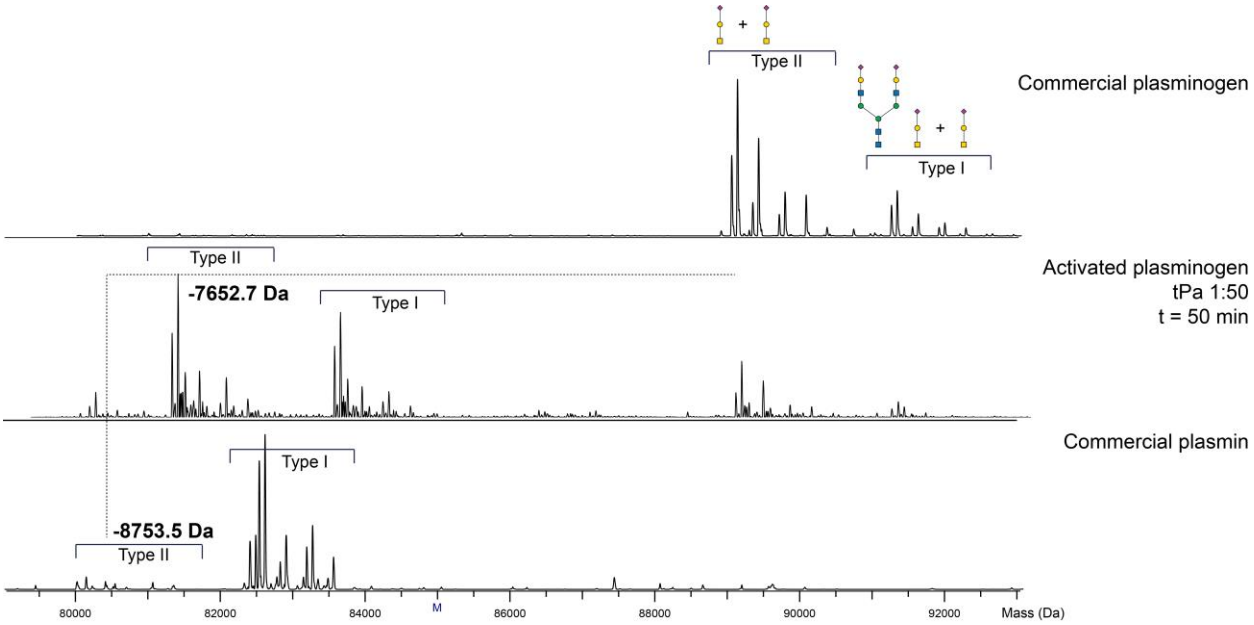


Supplementary figure 8: peptide analysis of low mass range of native MS Plg activation reveals the two-step N-terminal cleavage is always initiated by cleaving AA 1-68 first. Subsequent cleavage products of the activation peptide are also observed. Inlay: magnified view of the [1-68] peptide, overlaid with the simulated isotopic envelope of [1-68] with both disulfide bonds intact. An excellent agreement is observed.

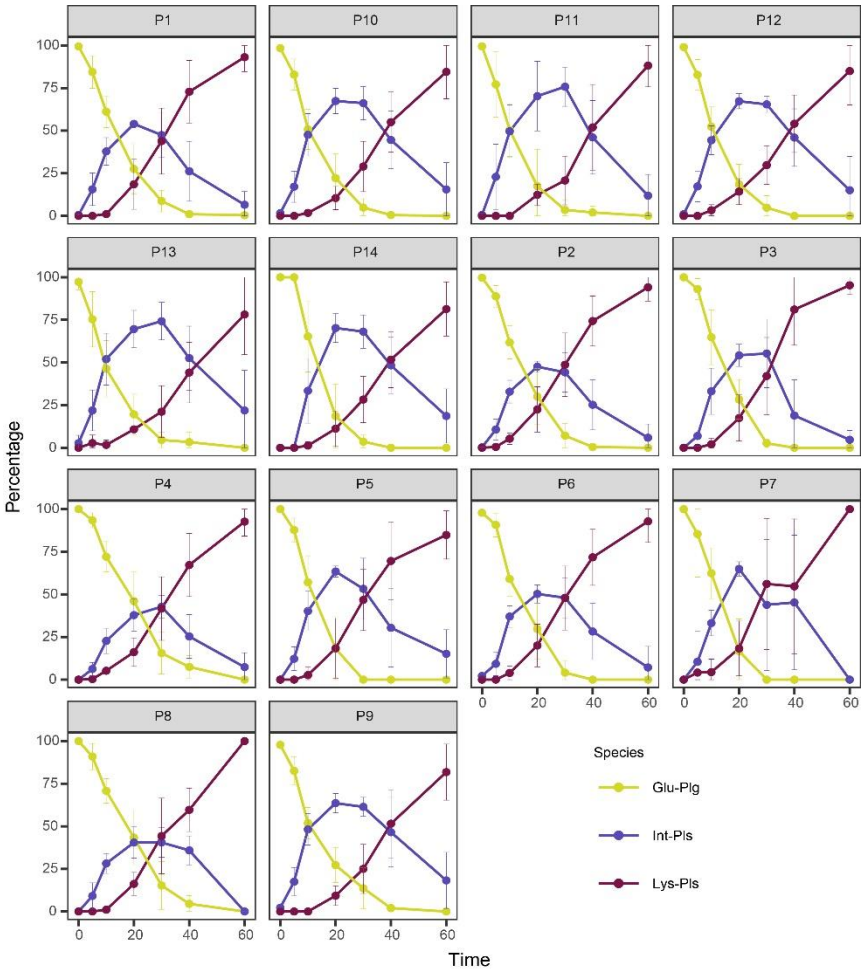
A.



B.



Supplementary figure 9: A. activation using a bacterial kinase (SAK) follows the same two-step N-terminal cleavage as activation with uPa. Rates of activation at a molar ratio of 1:500 (Plg:SAK) are comparable. **B**. activation with tPa similarly shows a two-step N-terminal cleavage. At a 1:50 Plg:tPa ratio, activation appears somewhat slower than compared to uPa/SAK.



Supplementary figure 10: The two-step conversion of Plg to Plm can be monitored on a proteoform level by native MS. After addition of uPa, native mass spectra are recorded at different time intervals. The relative abundances of each proteoform from Glu-Plg, Int-Plm and Lys-Plm are plotted.

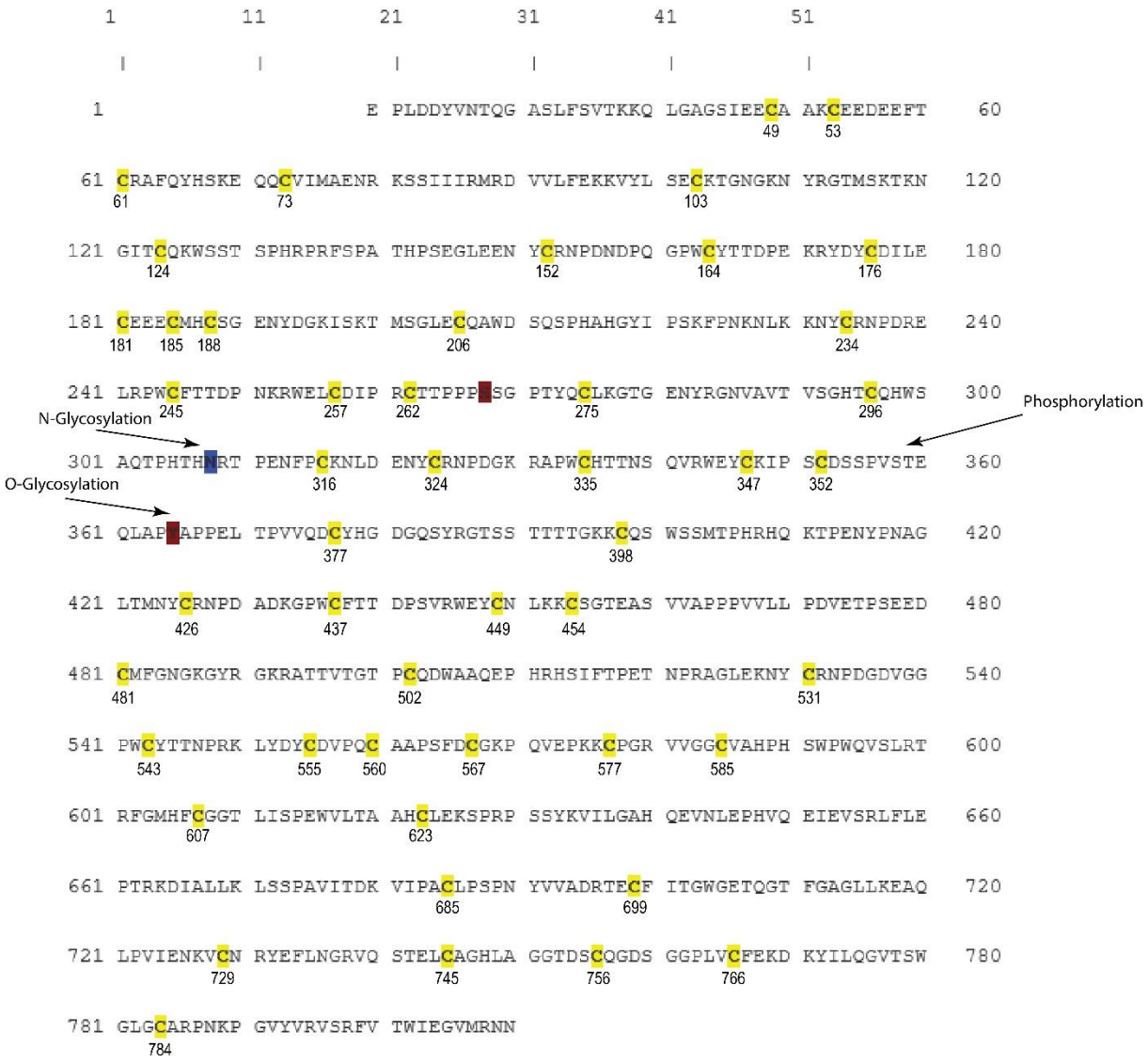
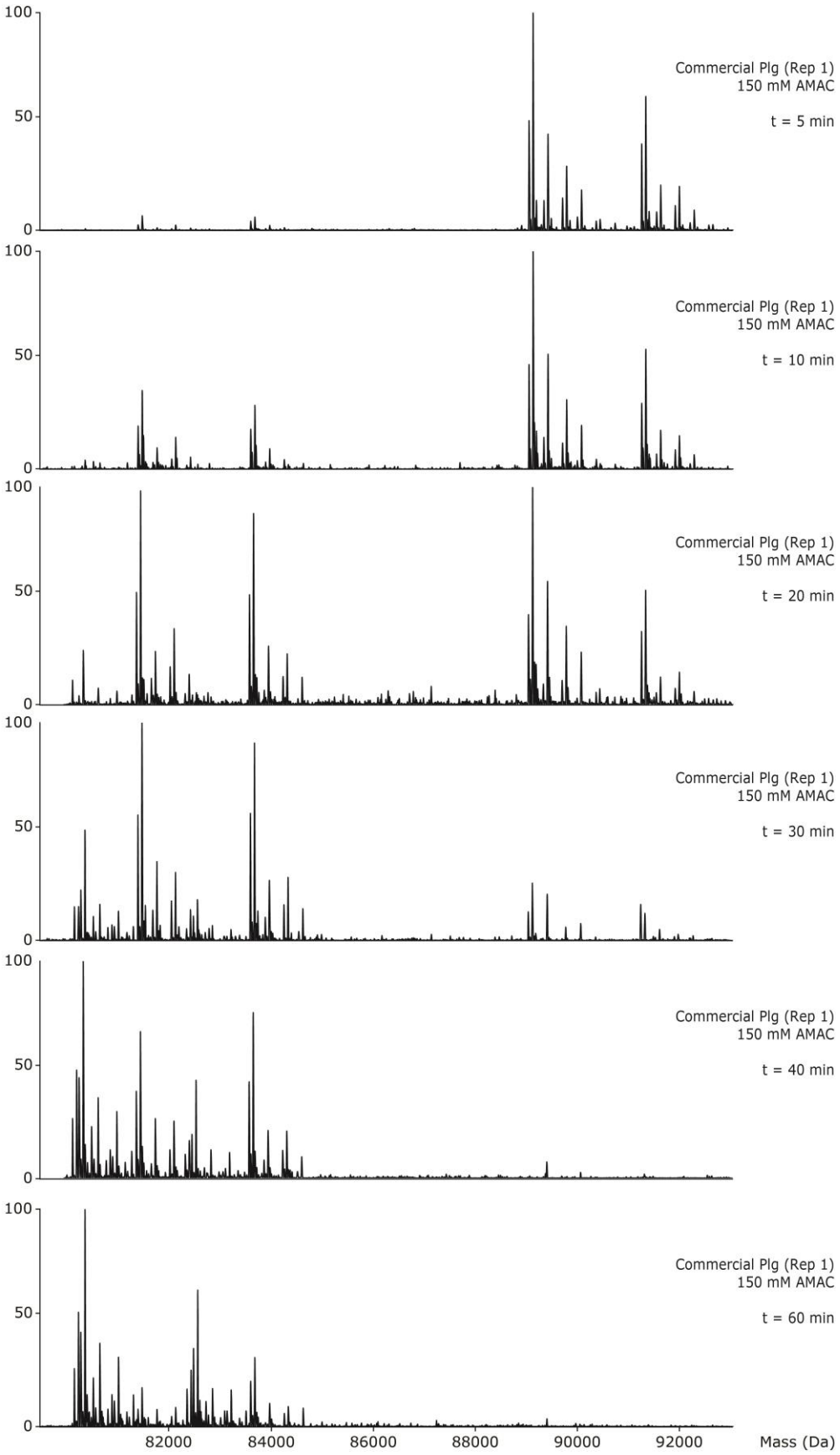


Table of disulfide bonds

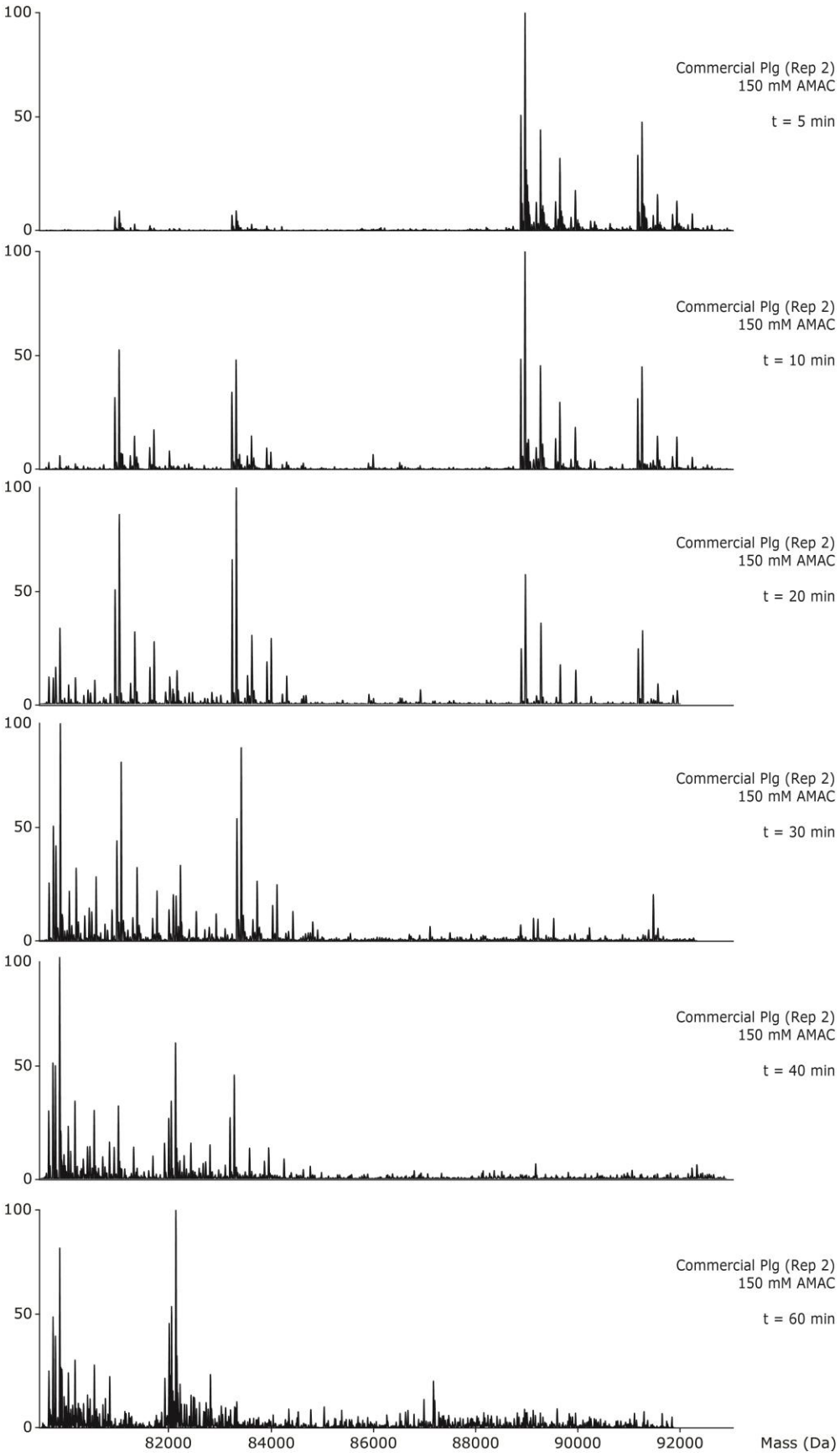
49	↔	73	234	↔	257	502	↔	543
53	↔	61	275	↔	352	531	↔	555
103	↔	181	296	↔	335	567	↔	685
124	↔	164	324	↔	347	577	↔	585
152	↔	176	377	↔	454	607	↔	623
185	↔	262	398	↔	437	699	↔	766
188	↔	316	426	↔	449	729	↔	745
206	↔	245	481	↔	560	756	↔	784

Supplementary figure 11: Sequence of human Plg. Cysteines in yellow are involved in disulfide bonds. Glycosylation and phosphorylation are indicated. A table provides the specific cysteines involved in each disulfide bond.

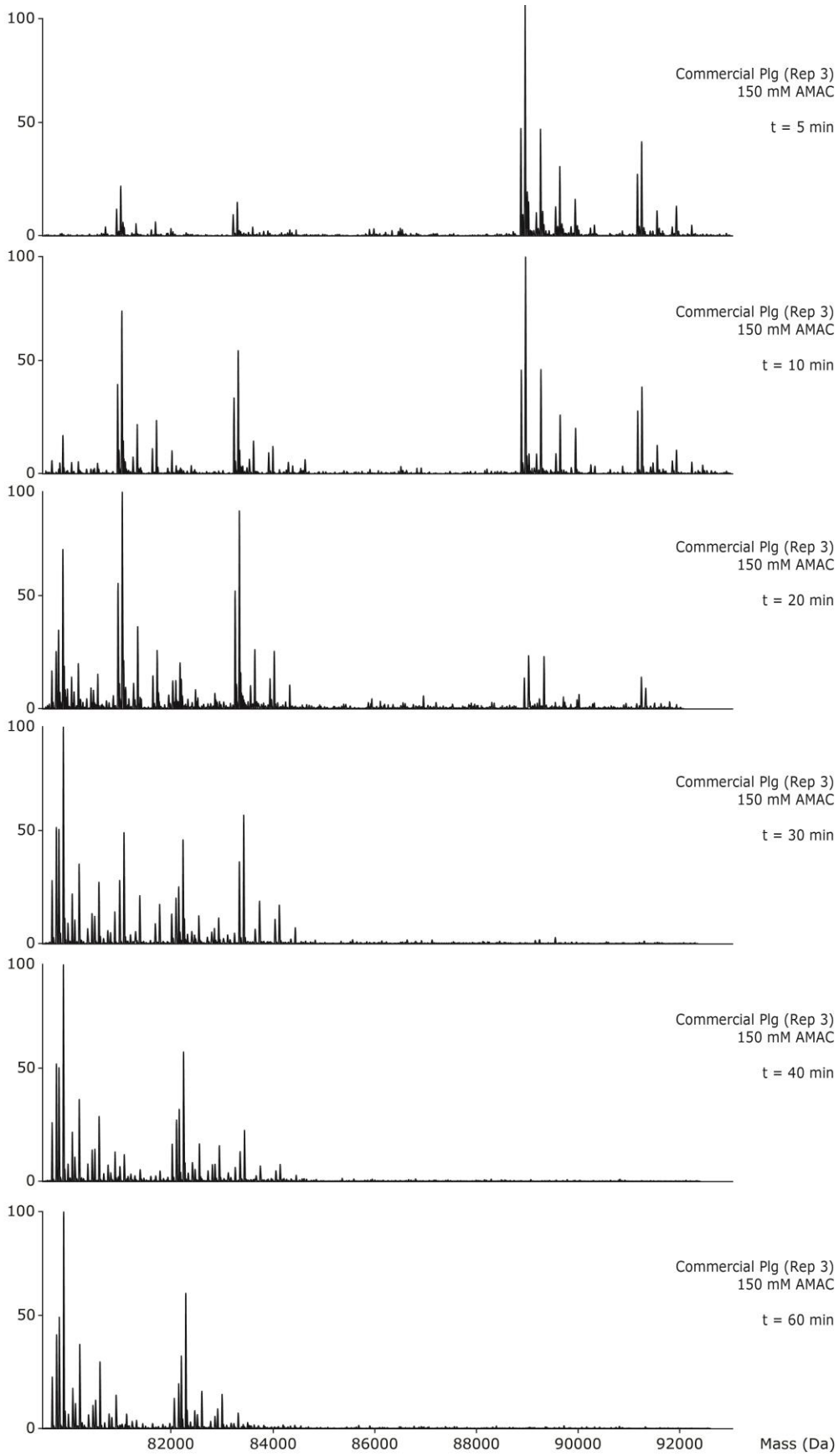
A


100
50
0
Commercial Plg (Rep 1)
150 mM AMAC
t = 5 min
Commercial Plg (Rep 1)
150 mM AMAC
t = 10 min
Commercial Plg (Rep 1)
150 mM AMAC
t = 20 min
Commercial Plg (Rep 1)
150 mM AMAC
t = 30 min
Commercial Plg (Rep 1)
150 mM AMAC
t = 40 min
Commercial Plg (Rep 1)
150 mM AMAC
t = 60 min
82000
84000
86000
88000
90000
92000
Mass (Da)

B


100
50
0
Commercial Plg (Rep 2)
150 mM AMAC
t = 5 min
Commercial Plg (Rep 2)
150 mM AMAC
t = 10 min
Commercial Plg (Rep 2)
150 mM AMAC
t = 20 min
Commercial Plg (Rep 2)
150 mM AMAC
t = 30 min
Commercial Plg (Rep 2)
150 mM AMAC
t = 40 min
Commercial Plg (Rep 2)
150 mM AMAC
t = 60 min
82000
84000
86000
88000
90000
92000
Mass (Da)

C


100
50
0
Commercial Plg (Rep 3)
150 mM AMAC
t = 5 min
Commercial Plg (Rep 3)
150 mM AMAC
t = 10 min
Commercial Plg (Rep 3)
150 mM AMAC
t = 20 min
Commercial Plg (Rep 3)
150 mM AMAC
t = 30 min
Commercial Plg (Rep 3)
150 mM AMAC
t = 40 min
Commercial Plg (Rep 3)
150 mM AMAC
t = 60 min
82000
84000
86000
88000
90000
92000
Mass (Da)

Supplementary figure 12: Native spectra of Plg activated by uPa over time showing **A**. technical replicate 1, **B**. technical replicate 2 and **C**. technical replicate 3.

Script 1: Calculation of sequence consensus in R using a FASTA file containing the selected Plg genes.

```
library(msa)

#load the Fasta file for sequence alignment
plgalignfile <- "F:/top250PLG.fasta"
plgalign <- readAAStringSet(plgalignfile)

#alignment processing
plgalign1 <- msa(plgalign)

#to view the alignment
print(plgalign1, show="complete")
view(plgalign1)

#create a consensus matrix based on the aligned amino acid number
#a range is selected by [1, 3]. a single AA is selected by [, 3]
#the example below, of 1190, shows the consensus matrix for O glycosylation site S249
conMat <- consensusMatrix(plgalign1)
dim(conMat)
conMat[, 1190]
```

Script 2: Simulation of a native MS spectrum from bottom-up data. The PTM table consists of a column P1-P14, a column indicating which PTM is present (text) and a column with the relative abundance of that PTM.

```
library(stringr)

library(dplyr)

library(readxl)

library(data.table)

library(readr)


#select annotations and import glycan masses

PTMs <- read_excel("/PTM.xlsx")

glycounitMass <- c("N"=203.1950, "H"=162.1424, "P"=79.9799,"S"=291.2579)

PTMs <- read_excel("/PTM.xlsx")%>%

mutate(N=as.numeric(str_extract(str_extract(PTMs$PTM,"N\\d"),"\\d"))*glycounitMass["N"
],

H=as.numeric(str_extract(str_extract(PTMs$PTM,"H\\d"),"\\d"))*glycounitMass["H"],

P=as.numeric(str_extract(str_extract(PTMs$PTM,"P\\d"),"\\d"))*glycounitMass["P"],

S=as.numeric(str_extract(str_extract(PTMs$PTM,"S\\d"),"\\d"))*glycounitMass["S"],)

PTMs$PTMmass<-rowSums(PTMs[,-(1:3)],na.rm=TRUE)


#Backbone mass corrected for disulfide bridges and fixed modifications

backbone<-88384.4


#Combinations of different site-specific information

sites_S01<-split(PTMs,PTMs$Site) #Split dataframe by the number of modified sites


#All possible combinations of modified sites

Calsites_S01<-expand.grid(sites_S01[[1]]$PTM,

                          sites_S01[[2]]$PTM,

                          sites_S01[[3]]$PTM,

                          sites_S01[[4]]$PTM)
```

```
colnames(Calsites_S01)<-names(sites_S01)


#Calculate the theoretical molecular weight of protein with all possible combinations
of modifications
Calcombination_S01<-expand.grid(sites_S01[[1]]$PTMmass,

                                sites_S01[[2]]$PTMmass,

                                sites_S01[[3]]$PTMmass,

                                sites_S01[[4]]$PTMmass)



colnames(Calcombination_S01)<-paste(names(sites_S01),"_mass")
Calcombination_S01<-Calcombination_S01%>%

mutate(CalPTMmass=rowSums(Calcombination_S01,na.rm=TRUE),CalMW=CalPTMmass+backbone)
#Calculate possibility of proteoform with site-specific occupancies
Abundance_S01<-expand.grid(sites_S01[[1]]$RelativeAbundance,

                           sites_S01[[2]]$RelativeAbundance,

                           sites_S01[[3]]$RelativeAbundance,

                           sites_S01[[4]]$RelativeAbundance)


colnames(Abundance_S01)<-paste(names(sites_S01),"_abundance")
Abundance_S01<-Abundance_S01%>%
  mutate(totalAbundance=apply(Abundance_S01, 1, prod))


#Make the table with information of all possible site-specific modifications and their
possibilities
AllCombinations_S01<-data.frame(Calsites_S01,Calcombination_S01,Abundance_S01)


#Calculate the m/z of all theoretical proteoforms in certain charge states (choose the
most dominant charge states in native spectra)
```

```
AllCombinations_S01<-AllCombinations_S01%>%

  mutate(mz20=(CalMW+20*1.007276)/20,

         mz19=(CalMW+19*1.007276)/19,

         mz18=(CalMW+18*1.007276)/18,

         mz17=(CalMW+17*1.007276)/17,

         mz0=(CalMW+0*1.007276)/1)

write.csv(AllCombinations_S01,file='AllcombPlaminogen1.csv')

#Annonate peaks in experimental spectrum to the most possible PTM combination with
least difference within certain ppm

readfindmax<-function(rawdata,caldata,chargetouse,chargenumber,ppm){

  expdata<- read.csv(rawdata,sep='\t',col.names= c("mz","int"))

  maxint<-do.call(rbind, lapply(split(caldata,chargetouse), function(x)
{return(x[which.max(x$totalAbundance),])}))

  expmz<-expdata$mz
    c1 <- c()
    c2 <- c()

    for (i in chargetouse){

    dif <- expmz - i
    print(dif)

    ppmcut <- i * ppm/1000000
    print(ppmcut)
```

```
    d<- dif[abs(dif) <= ppmcut] + i # all m/z withtin certain ppm
    print(d)
    for(j in d){

      c1 <- append(c1, i)

      c2 <- append(c2, j)

    }

  }

  #print(c2)
  output<-
data.frame(maxint[match(c1,maxint[,chargenumber]),],expdata[match(c2,expdata$mz),])

  finaloutput<-do.call(rbind, lapply(split(output,output$mz), function(x)
{return(x[which.max(x$totalAbundance),])}))%>%

    mutate(relint=int/max(int)*100)

  ppmcal<-finaloutput[,chargenumber]

  finaloutput$Deltappm<-abs(finaloutput$mz/ppmcal-1)*1000000

  setnames(finaloutput,"mz",paste("expmz",chargenumber))

  setnames(finaloutput,"int",paste("int",chargenumber))

  setnames(finaloutput,"relint",paste("relint",chargenumber))

  setnames(finaloutput,"Deltappm",paste("Deltappm",chargenumber))
  }

write.csv(AllCombinations_S01,file='AllcombPlaminogen1.csv')
```

```
#Import experimental spectrum peak list (example can be found in the upload "S1.txt"
file with two columns: m/z, intensity)

S1<-readfindmax('C:/ ',AllCombinations_S01,AllCombinations_S01$CalMW,"CalMW",100)

rownames(S1)<-c(1:length(rownames(S1)))

write.csv(AllCombinations_S01,file='AllCombinations_S01.csv')


#simulated spectrum

generatePseudoGaussianSpectrum <- function(x, y, sd, xlim=range(x)){

  stopifnot(length(x) == length(y))

  plot_x <- seq(xlim[1], xlim[2], by=1)

  ans_y <- numeric(length(plot_x))

  for(i in 1:length(x)){

    plot_y <- dnorm(plot_x, mean=x[i], sd=sd)

    plot_y <- y[i] / max(plot_y) * plot_y

    ans_y <- pmax(ans_y, plot_y, na.rm=TRUE)

  }

  return(data.frame(x=plot_x, y=ans_y))

}

toPlot <- generatePseudoGaussianSpectrum(

  x=as.numeric(AllCombinations_S01[,20]),

  y=100/max(AllCombinations_S01[,15])*AllCombinations_S01[,15],

  sd=5,

  xlim=c(88500, 92300))
```