Using high-throughput computing to predict future lapses back to alcohol use

Kendra Wyant

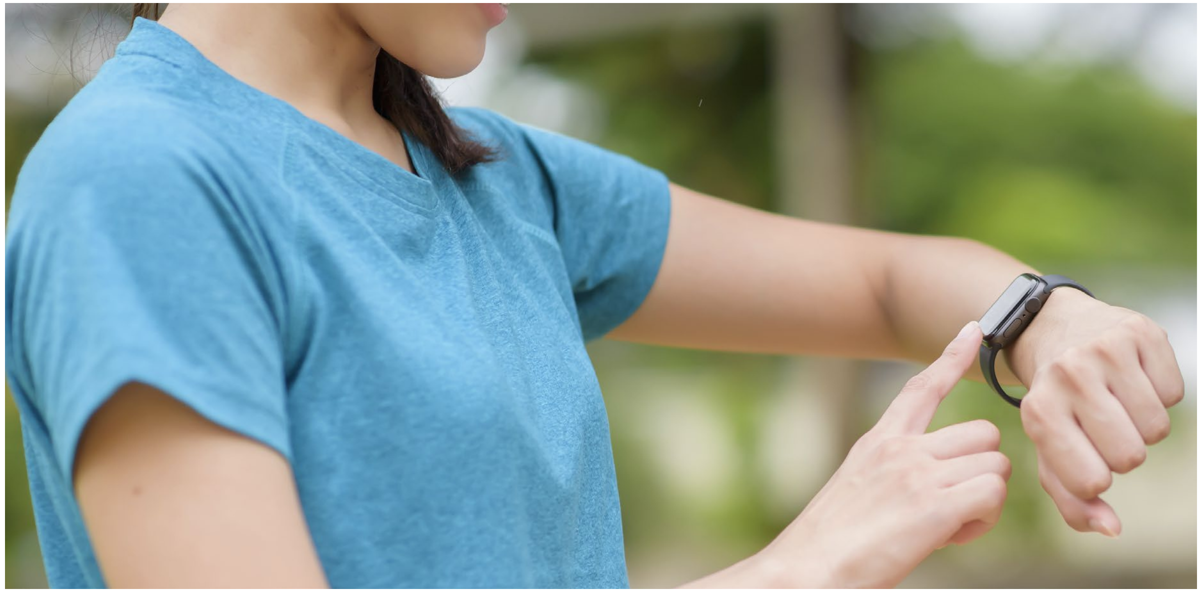
PI: John Curtin

PSYCHOLOGY

Addiction Research Center
UNIVERSITY OF WISCONSIN – MADISON

CHTC

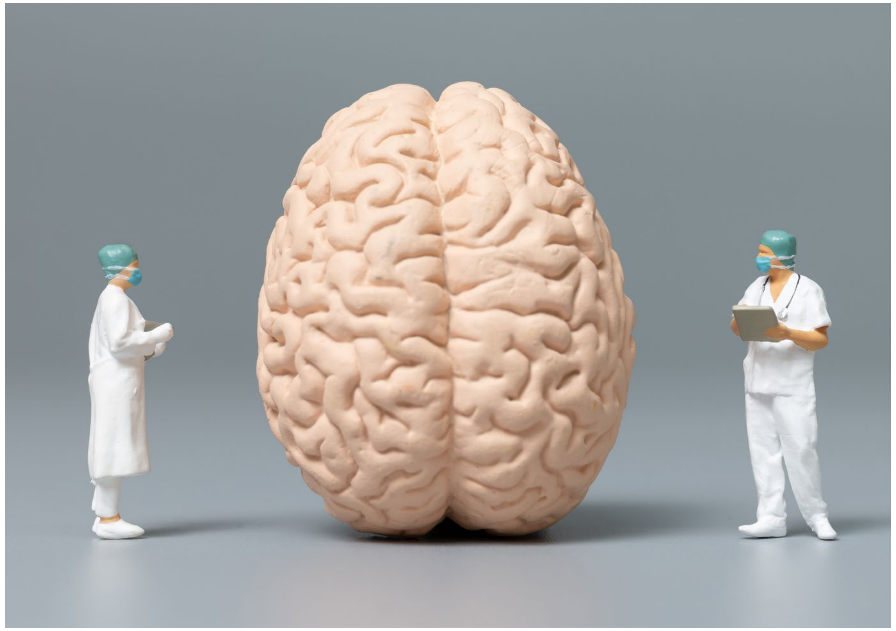
Open Science Grid



You
Tube
Insta

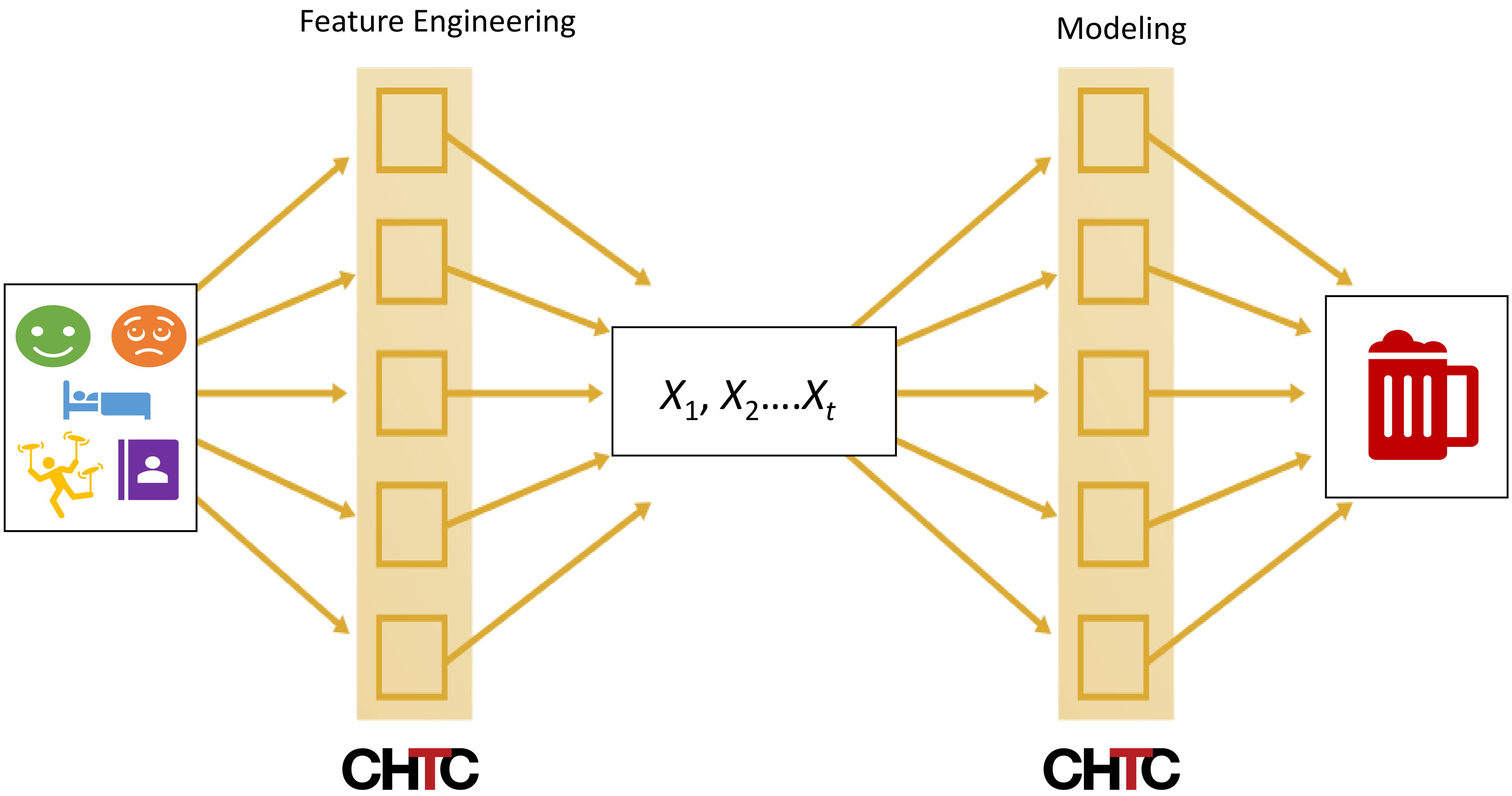
Personal Sensing and Mental Health

- Screening
  - Passive
  - Scalable
- Monitoring
  - Intervention prior to relapse



Alcohol Use Disorder

- AUD is a chronic relapsing disease
- Lapses are often early signs of relapse
- A temporally dynamic sensing system can capture day-to-day changes in lapse risk



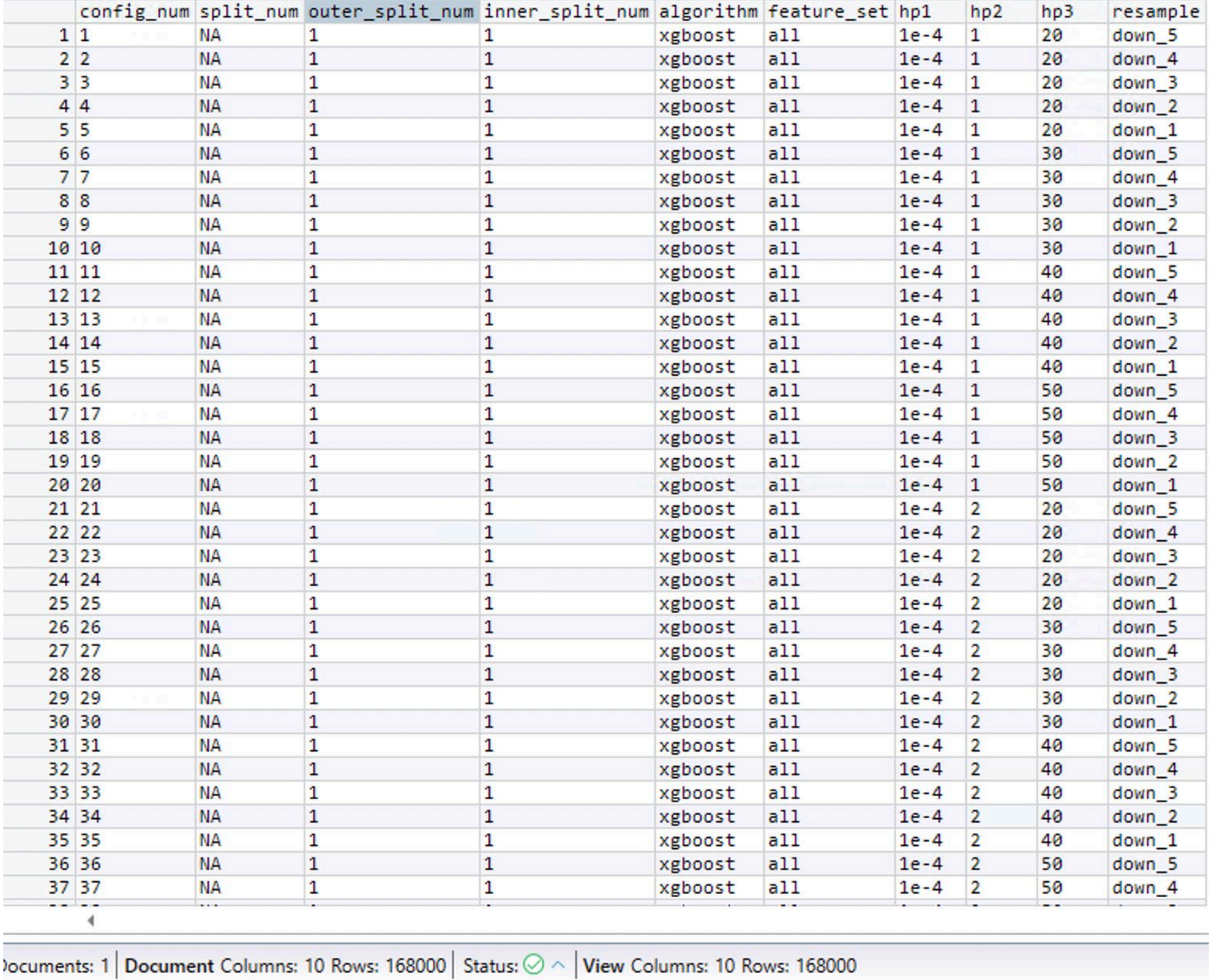
Feature Engineering

Modeling

$X_1, X_2....X_t$

Why CHTC ?

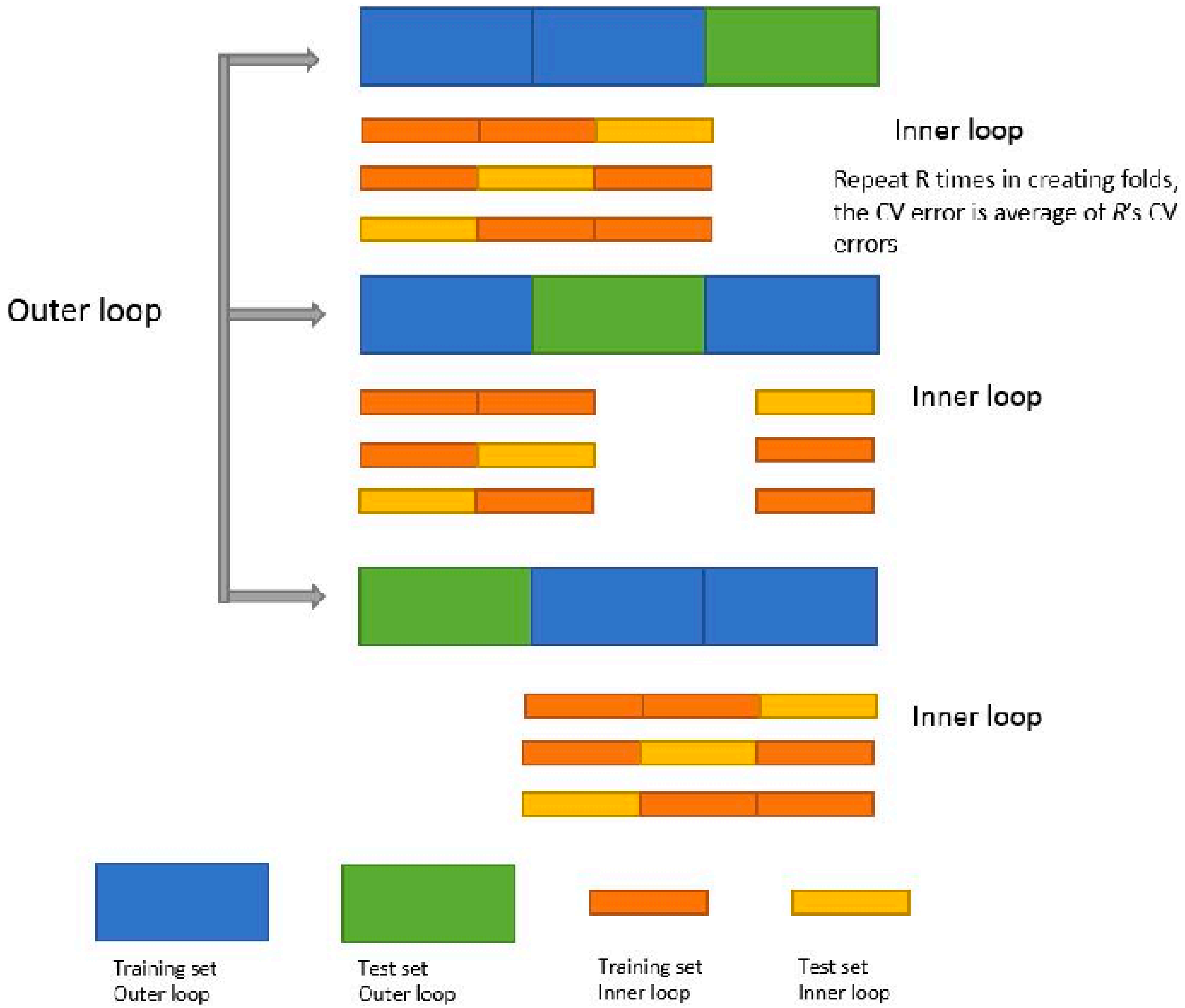
- Memory requirements
- Time



	config_num	split_num	outer_split_num	inner_split_num	algorithm	feature_set	hp1	hp2	hp3	resample
1	1	NA	1	1	xgboost	all	1e-4	1	20	down_5
2	2	NA	1	1	xgboost	all	1e-4	1	20	down_4
3	3	NA	1	1	xgboost	all	1e-4	1	20	down_3
4	4	NA	1	1	xgboost	all	1e-4	1	20	down_2
5	5	NA	1	1	xgboost	all	1e-4	1	20	down_1
6	6	NA	1	1	xgboost	all	1e-4	1	30	down_5
7	7	NA	1	1	xgboost	all	1e-4	1	30	down_4
8	8	NA	1	1	xgboost	all	1e-4	1	30	down_3
9	9	NA	1	1	xgboost	all	1e-4	1	30	down_2
10	10	NA	1	1	xgboost	all	1e-4	1	30	down_1
11	11	NA	1	1	xgboost	all	1e-4	1	40	down_5
12	12	NA	1	1	xgboost	all	1e-4	1	40	down_4
13	13	NA	1	1	xgboost	all	1e-4	1	40	down_3
14	14	NA	1	1	xgboost	all	1e-4	1	40	down_2
15	15	NA	1	1	xgboost	all	1e-4	1	40	down_1
16	16	NA	1	1	xgboost	all	1e-4	1	50	down_5
17	17	NA	1	1	xgboost	all	1e-4	1	50	down_4
18	18	NA	1	1	xgboost	all	1e-4	1	50	down_3
19	19	NA	1	1	xgboost	all	1e-4	1	50	down_2
20	20	NA	1	1	xgboost	all	1e-4	1	50	down_1
21	21	NA	1	1	xgboost	all	1e-4	2	20	down_5
22	22	NA	1	1	xgboost	all	1e-4	2	20	down_4
23	23	NA	1	1	xgboost	all	1e-4	2	20	down_3
24	24	NA	1	1	xgboost	all	1e-4	2	20	down_2
25	25	NA	1	1	xgboost	all	1e-4	2	20	down_1
26	26	NA	1	1	xgboost	all	1e-4	2	30	down_5
27	27	NA	1	1	xgboost	all	1e-4	2	30	down_4
28	28	NA	1	1	xgboost	all	1e-4	2	30	down_3
29	29	NA	1	1	xgboost	all	1e-4	2	30	down_2
30	30	NA	1	1	xgboost	all	1e-4	2	30	down_1
31	31	NA	1	1	xgboost	all	1e-4	2	40	down_5
32	32	NA	1	1	xgboost	all	1e-4	2	40	down_4
33	33	NA	1	1	xgboost	all	1e-4	2	40	down_3
34	34	NA	1	1	xgboost	all	1e-4	2	40	down_2
35	35	NA	1	1	xgboost	all	1e-4	2	40	down_1
36	36	NA	1	1	xgboost	all	1e-4	2	50	down_5
37	37	NA	1	1	xgboost	all	1e-4	2	50	down_4

ocuments: 1 | **Document** Columns: 10 Rows: 168000 | Status: ✓ | **View** Columns: 10 Rows: 168000

Iteration 1	Test	Train	Train	Train	Train
Iteration 2	Train	Test	Train	Train	Train
Iteration 3	Train	Train	Test	Train	Train
Iteration 4	Train	Train	Train	Test	Train
Iteration 5	Train	Train	Train	Train	Test



Outer loop

Inner loop

Repeat R times in creating folds, the CV error is average of *R*'s CV errors

Inner loop

Inner loop

Training set
Outer loop

Test set
Outer loop

Training set
Inner loop

Test set
Inner loop

Optimizing jobs

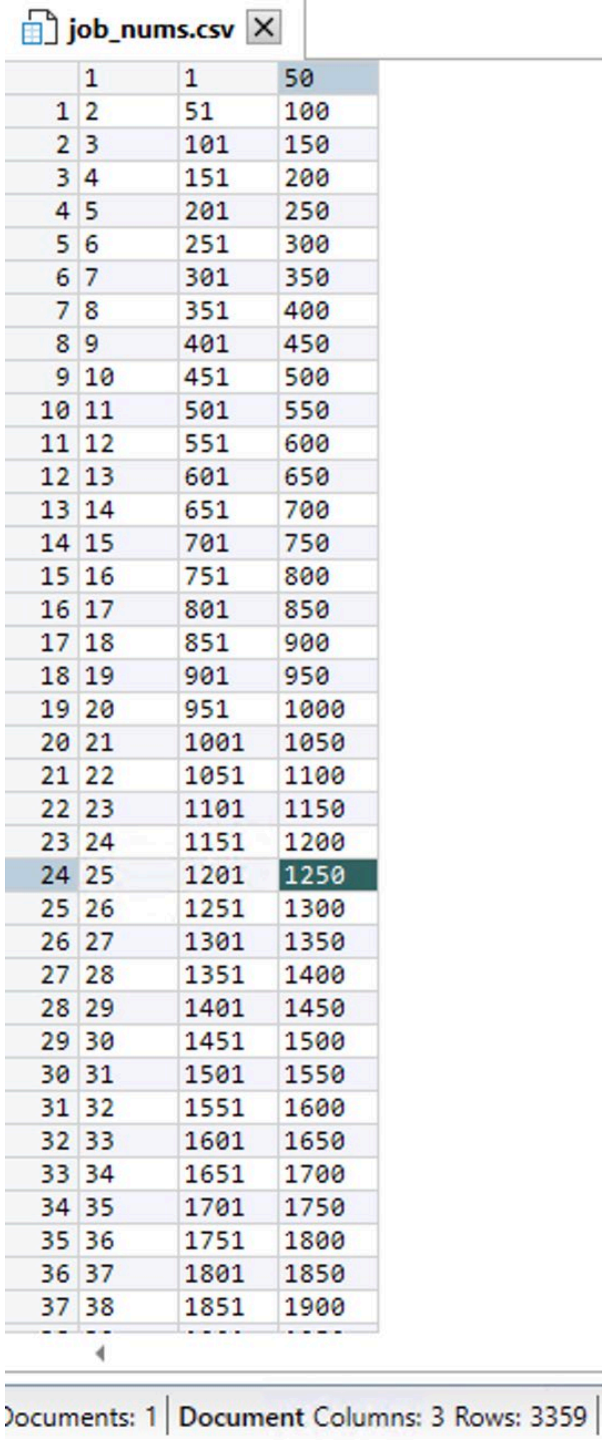
- Job run time
- Memory usage

```
condor_history $USER –limit 3 –af RequestMemory MemoryUsage RequestDisk DiskUsage
```

Optimizing jobs

- Job run time
- Memory usage

```
arguments = $(job_num) $(config_start) $(config_end)

queue job_num,config_start, config_end from job_nums.csv
```

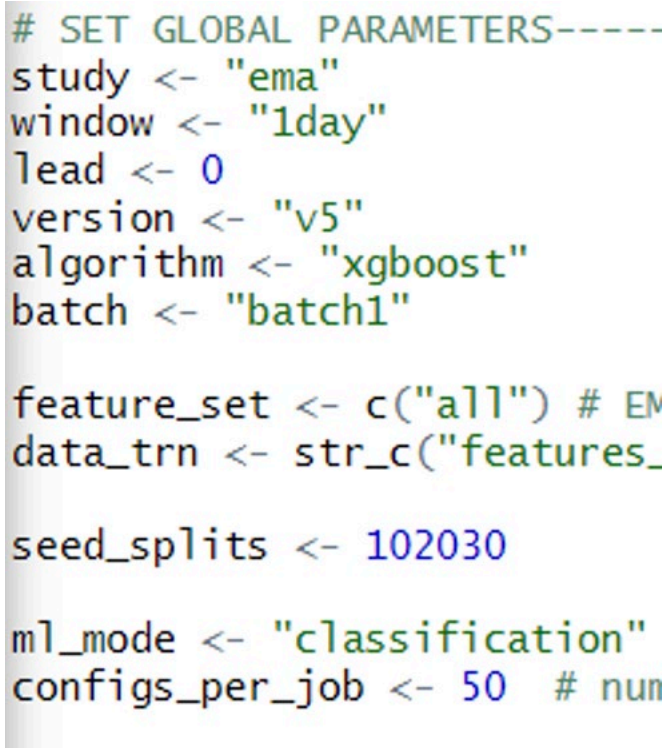


	1	1	50
1	2	51	100
2	3	101	150
3	4	151	200
4	5	201	250
5	6	251	300
6	7	301	350
7	8	351	400
8	9	401	450
9	10	451	500
10	11	501	550
11	12	551	600
12	13	601	650
13	14	651	700
14	15	701	750
15	16	751	800
16	17	801	850
17	18	851	900
18	19	901	950
19	20	951	1000
20	21	1001	1050
21	22	1051	1100
22	23	1101	1150
23	24	1151	1200
24	25	1201	1250
25	26	1251	1300
26	27	1301	1350
27	28	1351	1400
28	29	1401	1450
29	30	1451	1500
30	31	1501	1550
31	32	1551	1600
32	33	1601	1650
33	34	1651	1700
34	35	1701	1750
35	36	1751	1800
36	37	1801	1850
37	38	1851	1900

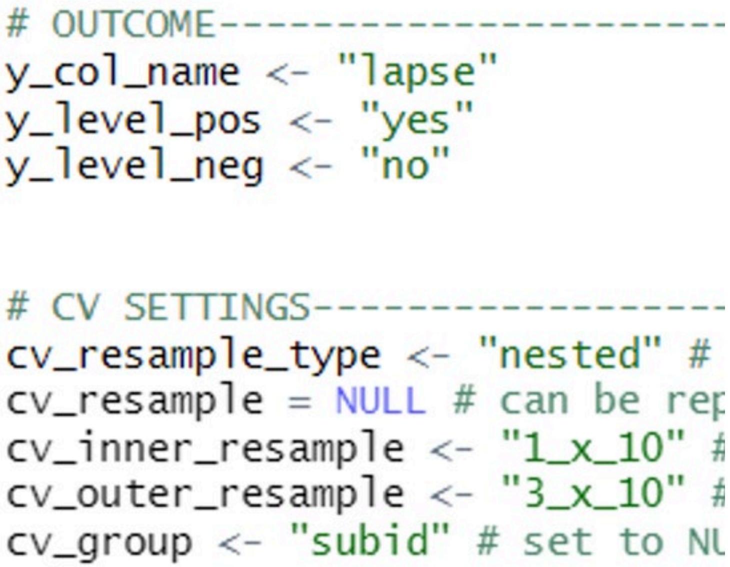
)ocuments: 1 | Document Columns: 3 Rows: 3359

Creating a generalized workflow

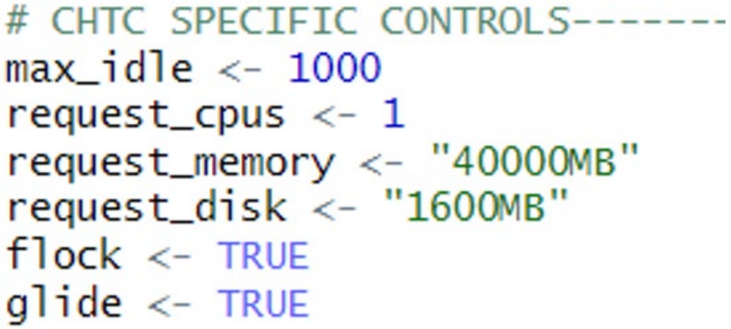
- Workflow – templates, functions
- Generalized to work with multiple data streams



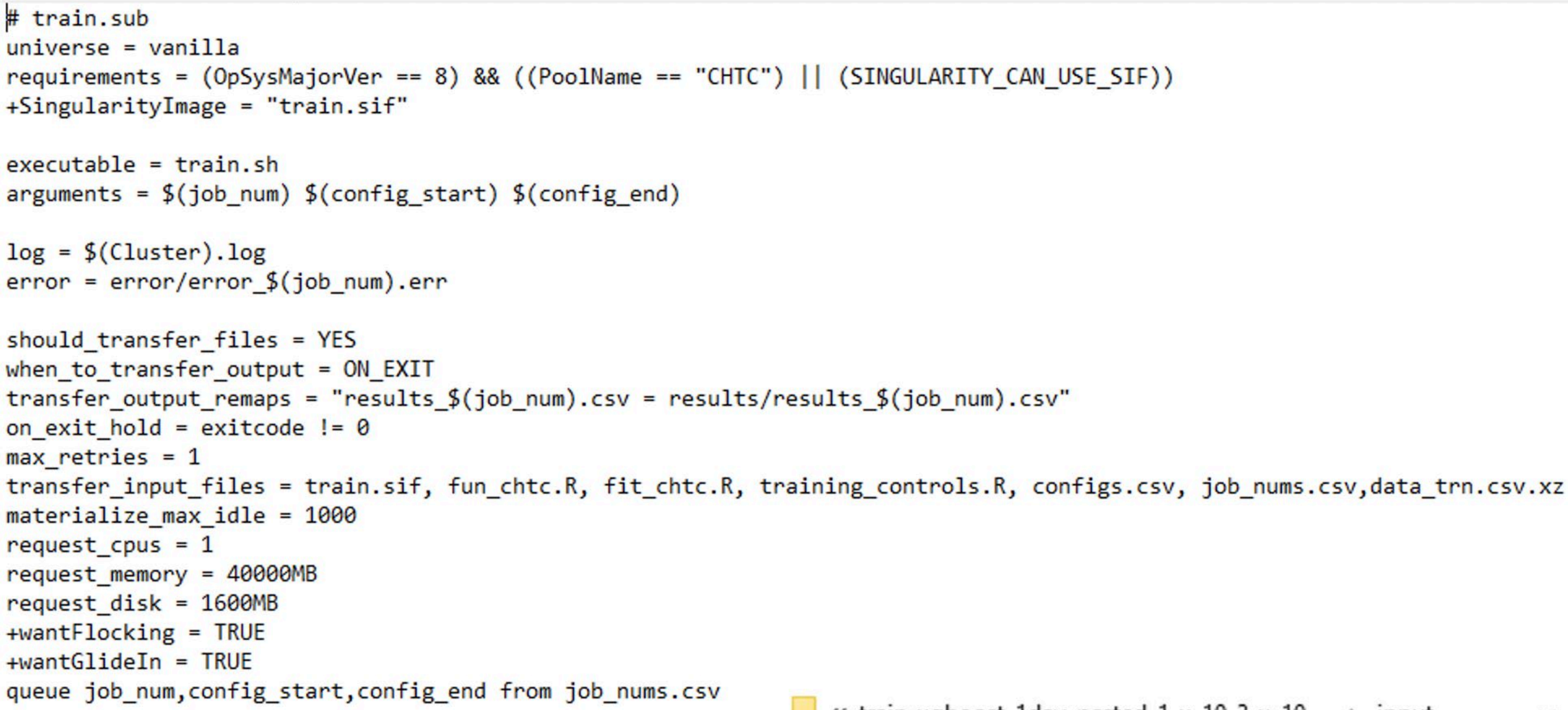
```
# SET GLOBAL PARAMETERS-----
study <- "ema"
window <- "1day"
lead <- 0
version <- "v5"
algorithm <- "xgboost"
batch <- "batch1"

feature_set <- c("all") # EM
data_trn <- str_c("features_

seed_splits <- 102030

ml_mode <- "classification"
configs_per_job <- 50  # num
```



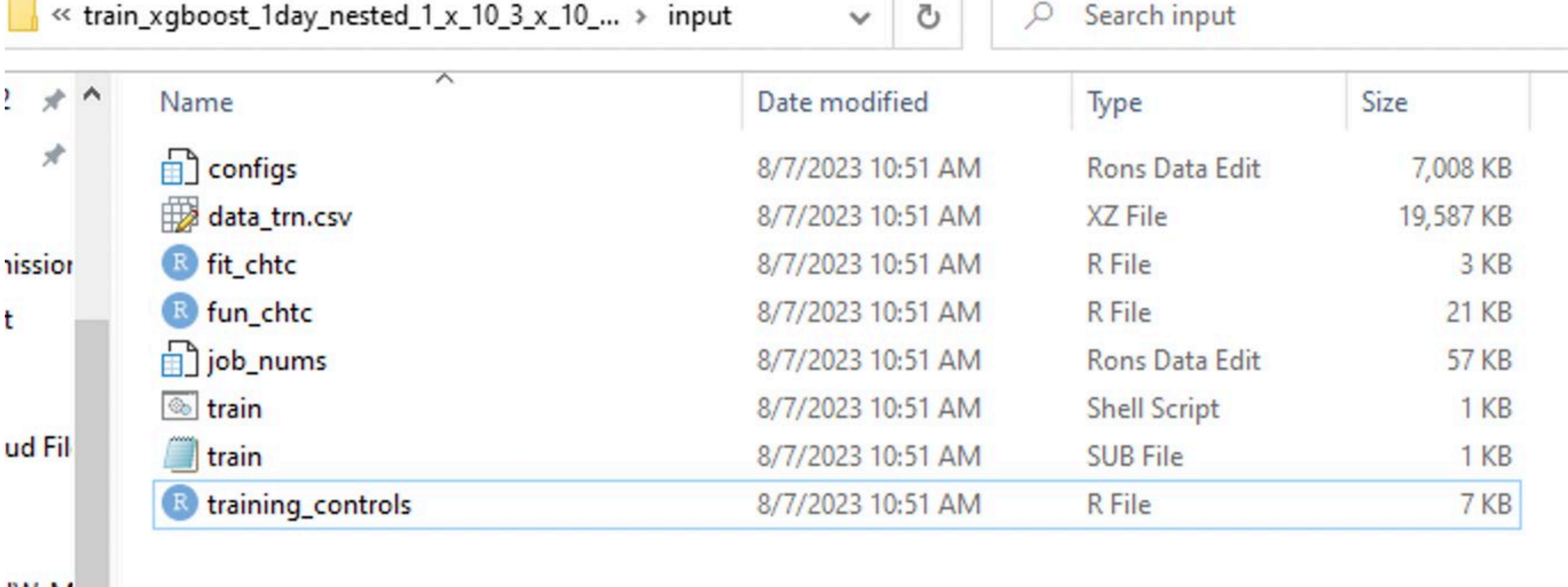
```
# OUTCOME----------------------
y_col_name <- "lapse"
y_level_pos <- "yes"
y_level_neg <- "no"


# CV SETTINGS-----------------
cv_resample_type <- "nested" #
cv_resample = NULL # can be rep
cv_inner_resample <- "1_x_10" #
cv_outer_resample <- "3_x_10" #
cv_group <- "subid" # set to NU
```



```
# CHTC SPECIFIC CONTROLS-------
max_idle <- 1000
request_cpus <- 1
request_memory <- "40000MB"
request_disk <- "1600MB"
flock <- TRUE
glide <- TRUE
```



```
# train.sub
universe = vanilla
requirements = (OpSysMajorVer == 8) && ((PoolName == "CHTC") || (SINGULARITY_CAN_USE_SIF))
+SingularityImage = "train.sif"

executable = train.sh
arguments = $(job_num) $(config_start) $(config_end)

log = $(Cluster).log
error = error/error_$(job_num).err

should_transfer_files = YES
when_to_transfer_output = ON_EXIT
transfer_output_remaps = "results_$(job_num).csv = results/results_$(job_num).csv"
on_exit_hold = exitcode != 0
max_retries = 1
transfer_input_files = train.sif, fun_chtc.R, fit_chtc.R, training_controls.R, configs.csv, job_nums.csv,data_trn.csv.xz
materialize_max_idle = 1000
request_cpus = 1
request_memory = 40000MB
request_disk = 1600MB
+wantFlocking = TRUE
+wantGlideIn = TRUE
queue job_num,config_start,config_end from job_nums.csv
```



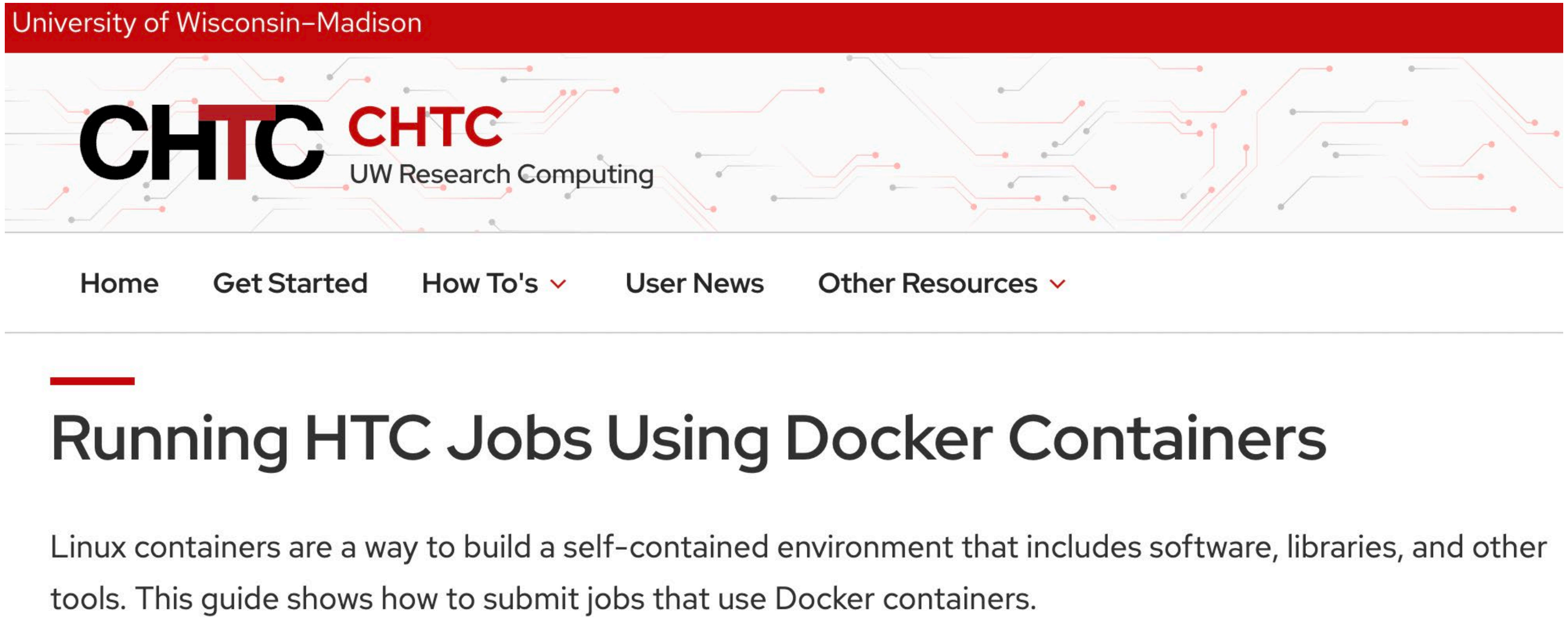
Name	Date modified	Type	Size
configs	8/7/2023 10:51 AM	Rons Data Edit	7,008 KB
data_trn.csv	8/7/2023 10:51 AM	XZ File	19,587 KB
fit_chtc	8/7/2023 10:51 AM	R File	3 KB
fun_chtc	8/7/2023 10:51 AM	R File	21 KB
job_nums	8/7/2023 10:51 AM	Rons Data Edit	57 KB
train	8/7/2023 10:51 AM	Shell Script	1 KB
train	8/7/2023 10:51 AM	SUB File	1 KB
training_controls	8/7/2023 10:51 AM	R File	7 KB

Troubleshooting

- Working with large files
    - Staging server
- Limited local CHTC machine matches
    - Flocking and gliding
    - Containers

Troubleshooting

- Working with large files
  - Staging server
- Limited local CHTC machine matches
  - Flocking and gliding
  - Containers



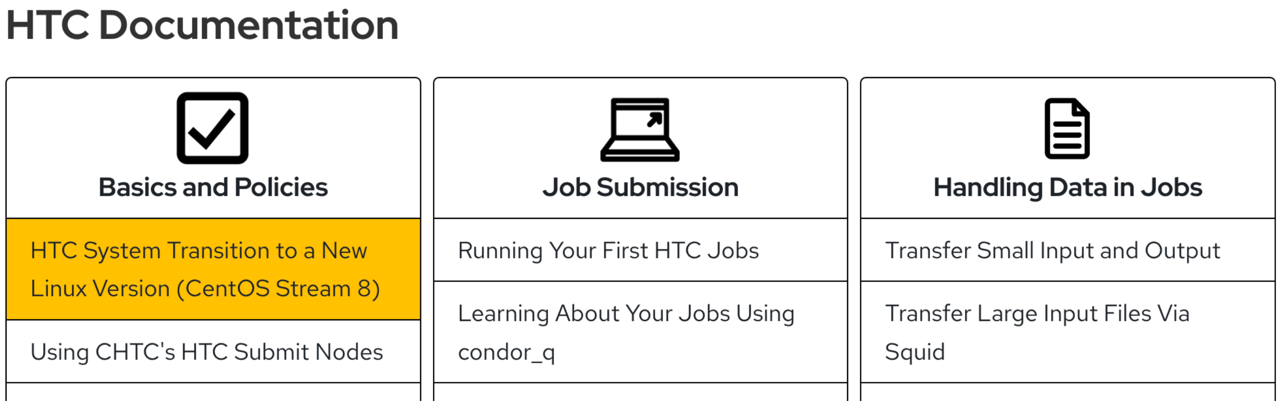
Running HTC Jobs Using Docker Containers

Linux containers are a way to build a self-contained environment that includes software, libraries, and other tools. This guide shows how to submit jobs that use Docker containers.

Helpful resources

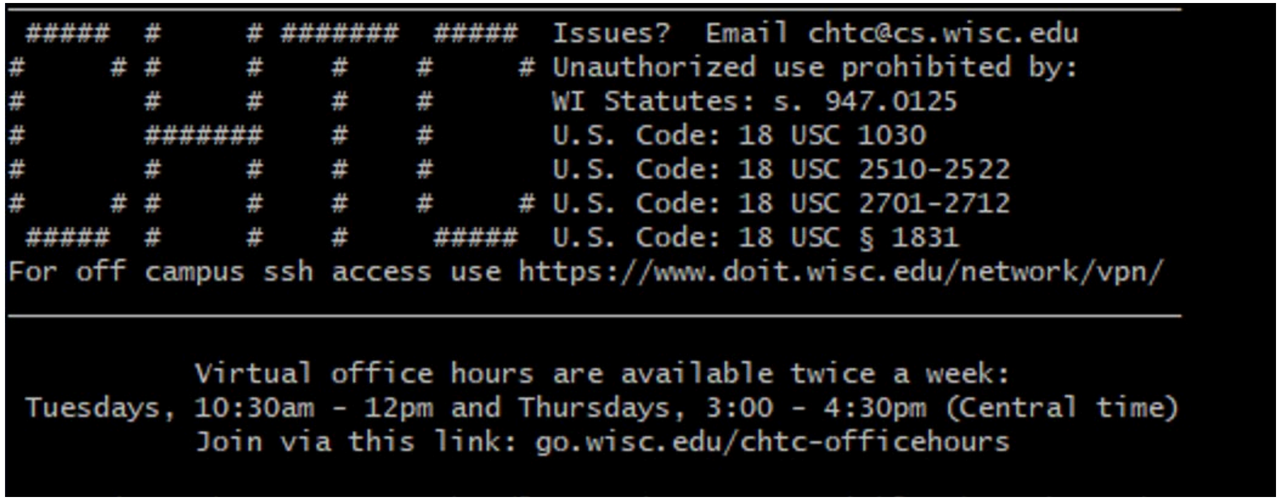
- Linux shell commands
- CHTC online guides
- CHTC office hours

freeCodeCamp(🔥)



HTC Documentation

Basics and Policies	Job Submission	Handling Data in Jobs
HTC System Transition to a New Linux Version (CentOS Stream 8)	Running Your First HTC Jobs	Transfer Small Input and Output
Using CHTC's HTC Submit Nodes	Learning About Your Jobs Using condor_q	Transfer Large Input Files Via Squid



```
 #####  #     # #######  #####  Issues?  Email chtc@cs.wisc.edu
#     # #     #    #    #     # Unauthorized use prohibited by:
#       #     #    #    #       WI Statutes: s. 947.0125
#       #######    #    #       U.S. Code: 18 USC 1030
#       #     #    #    #       U.S. Code: 18 USC 2510-2522
#     # #     #    #    #     # U.S. Code: 18 USC 2701-2712
 #####  #     #    #     #####  U.S. Code: 18 USC § 1831
For off campus ssh access use https://www.doit.wisc.edu/network/vpn/

           Virtual office hours are available twice a week:
 Tuesdays, 10:30am - 12pm and Thursdays, 3:00 - 4:30pm (Central time)
           Join via this link: go.wisc.edu/chtc-officehours
```