Wind Loads On High Rise Buildings

CEE 4770 Final Presentation

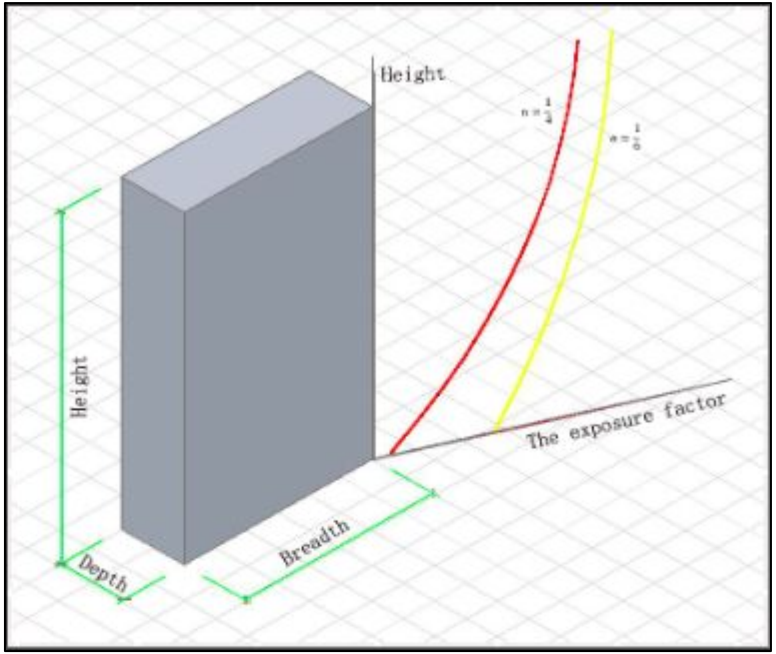
Alex Dzieman & Ananya Gangadhar

The goal is to model wind pressure coefficients on a high-rise building using data from wind tunnel experiments.

Plan of Action

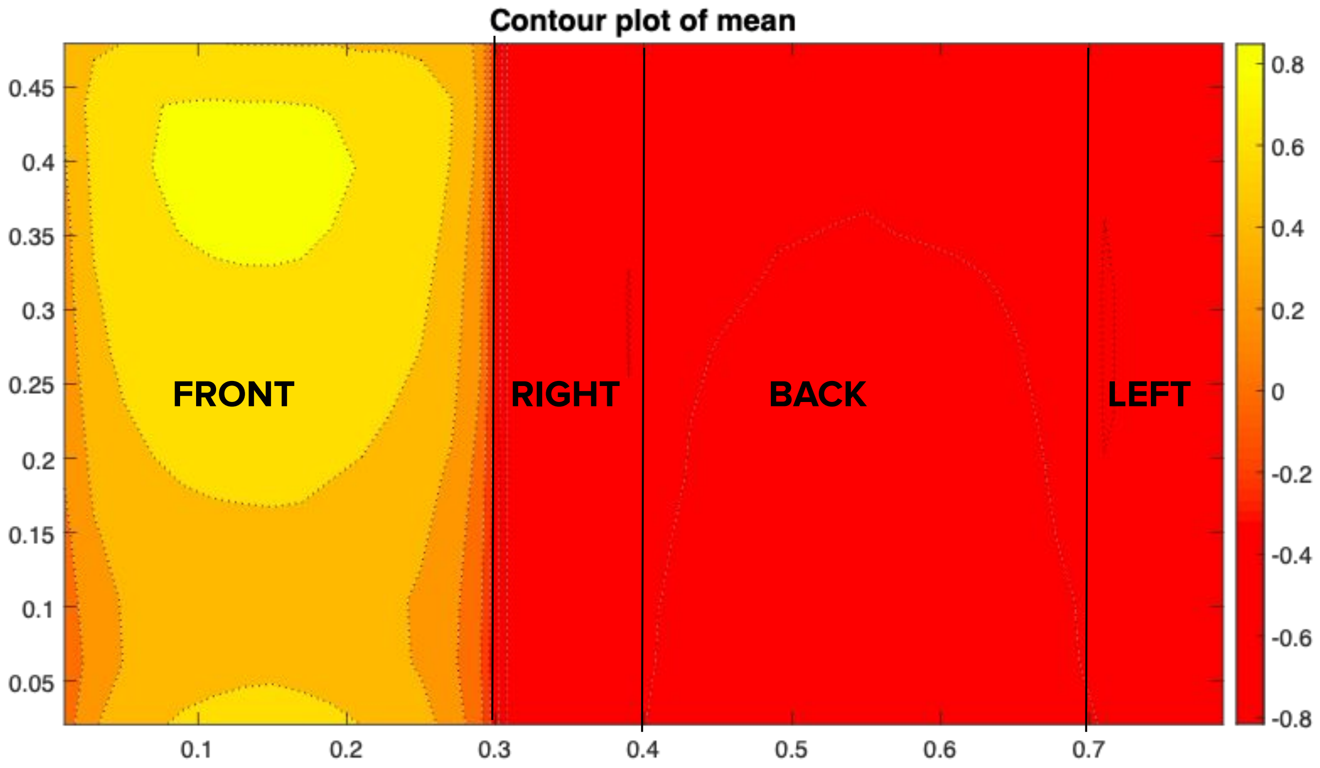
1. Derive statistical parameters of wind tunnel data
2. Make contour plots for raw data
3. Estimate distributions of wind tunnel data
4. Create a model for wind loads
5. Generate wind load samples and find maximum values
6. Estimate pressure coefficients for different exceedances
7. Create contour plots and draw conclusions

1. Derive statistical parameters from wind tunnel data

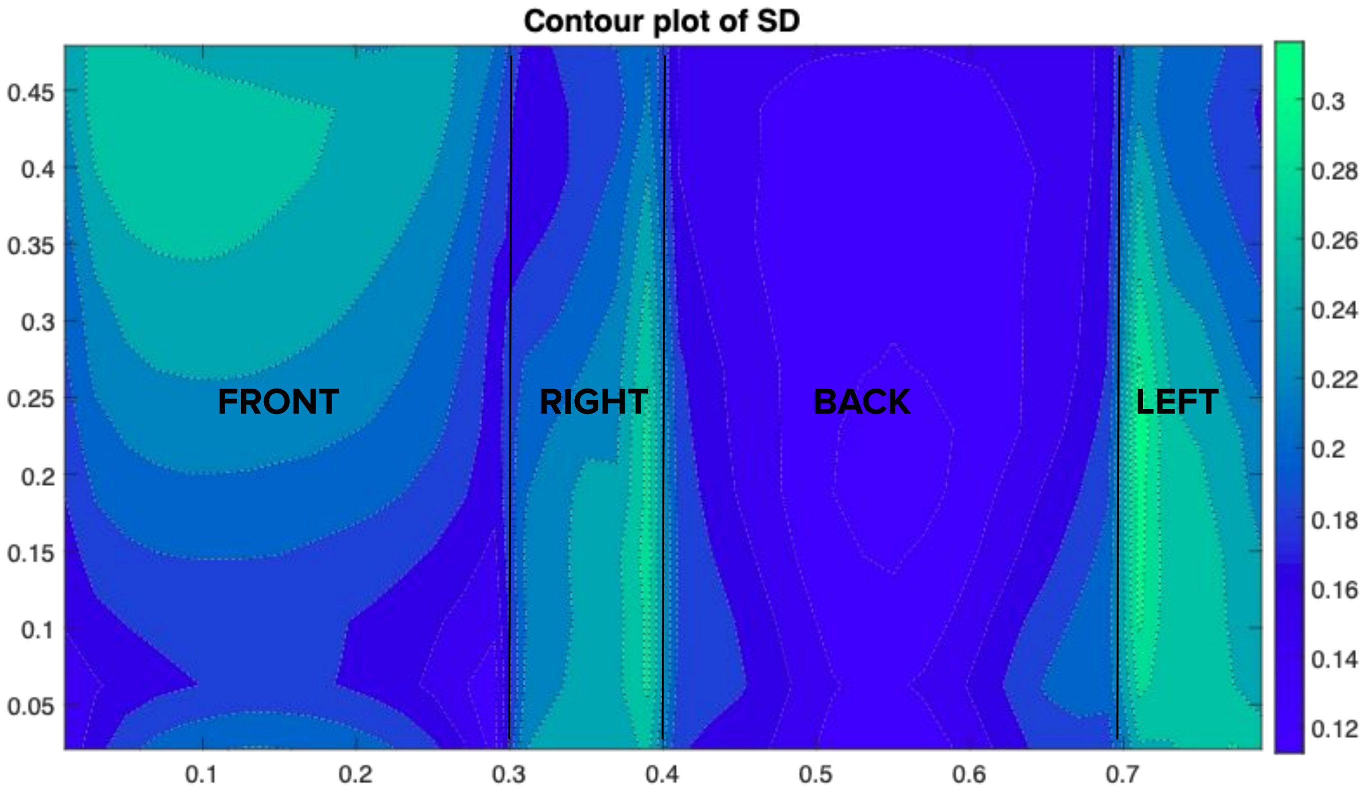


- Height:Breadth:Depth = 5:3:1
- Exposure factor = ¼
- Angle of wind exposure = 0°
- 480 pressure taps
- Time series data over ≈ 33 s
- Estimated mean, standard deviation, skewness and kurtosis

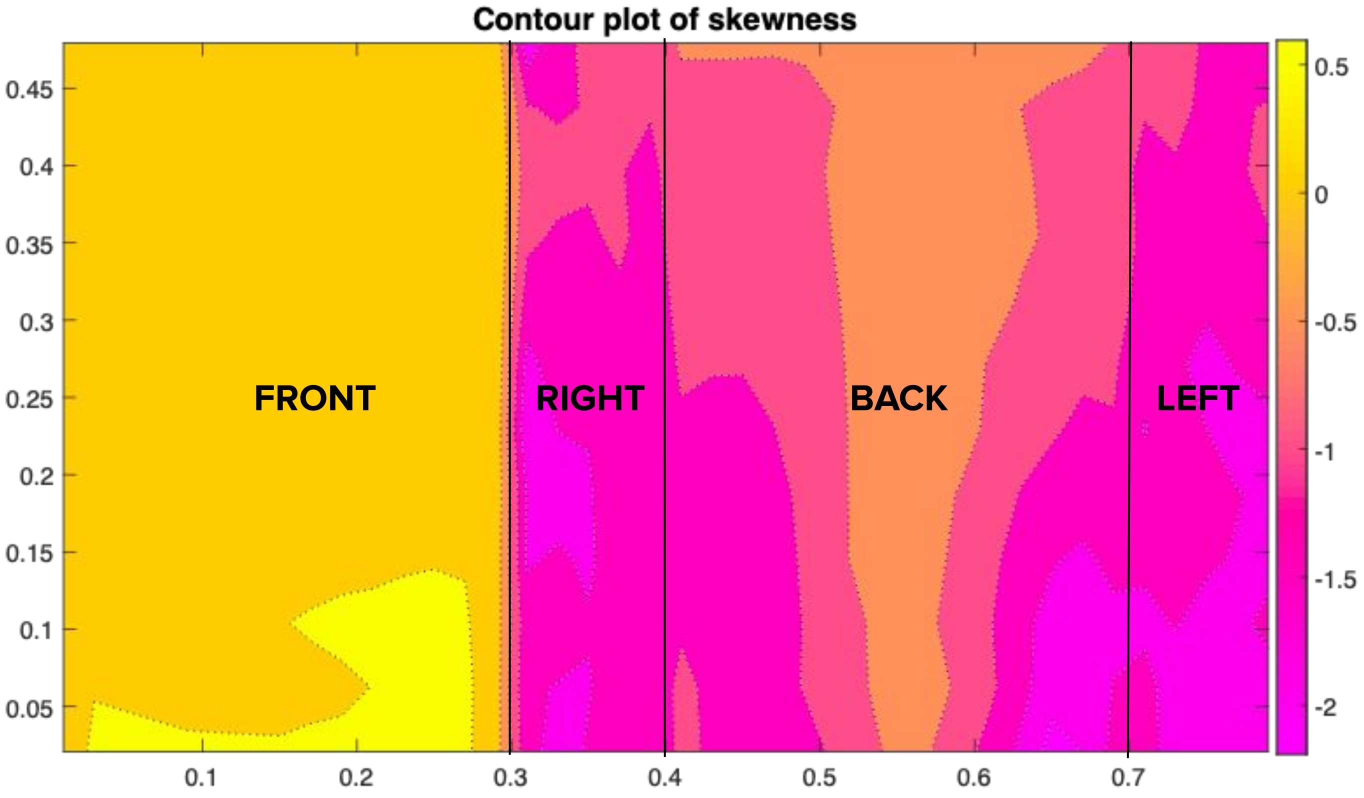
2. Make contour plots for raw data — Mean



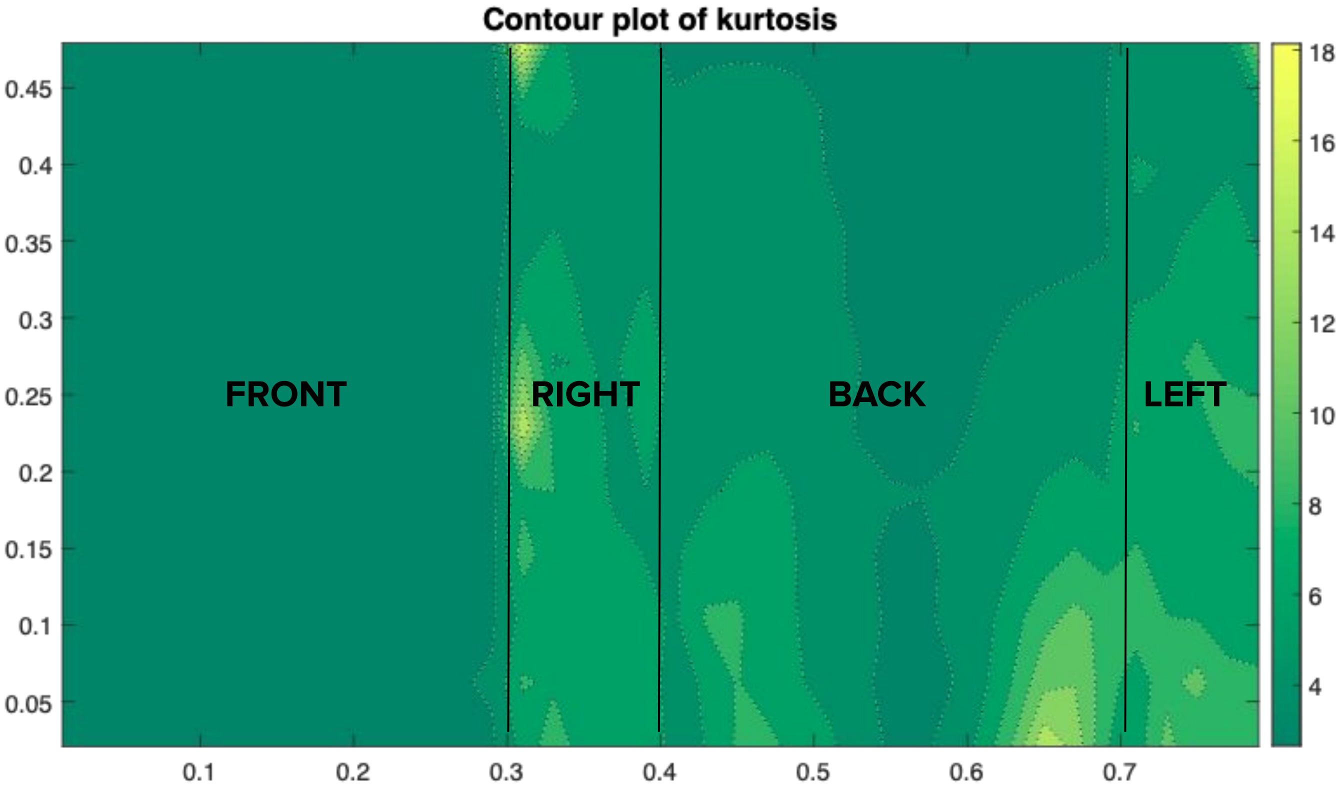
2. Make contour plots for raw data — SD



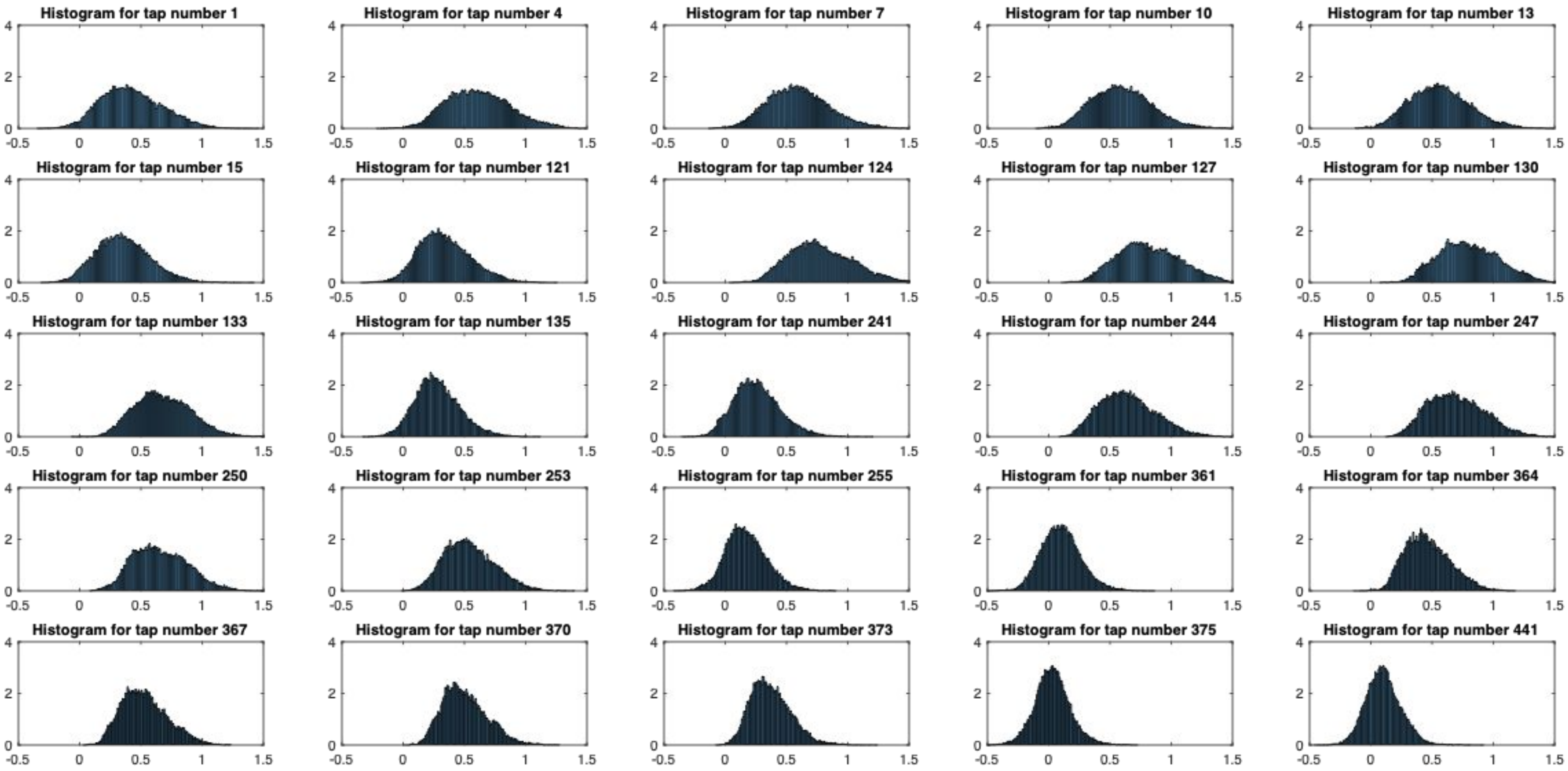
2. Make contour plots for raw data — Skewness



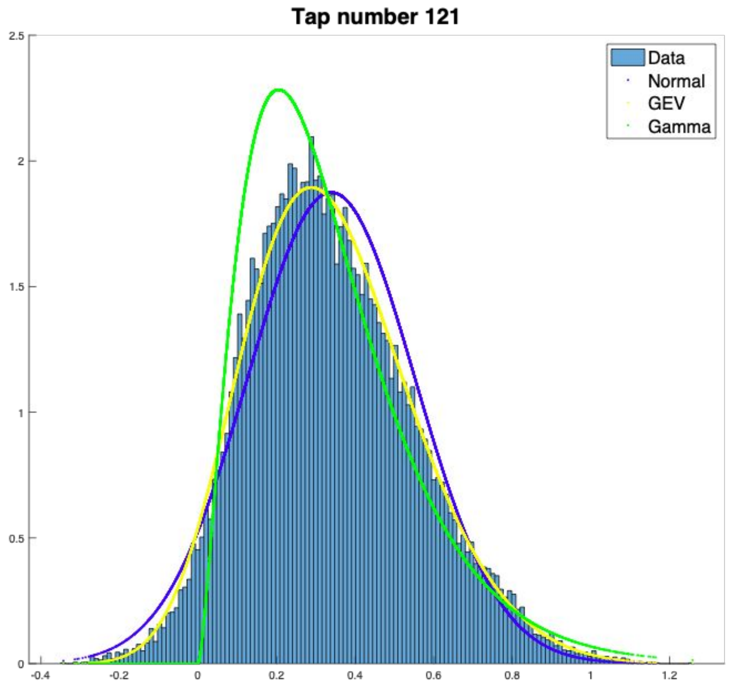
2. Make contour plots for raw data — Kurtosis



3. Estimate distributions of wind tunnel data

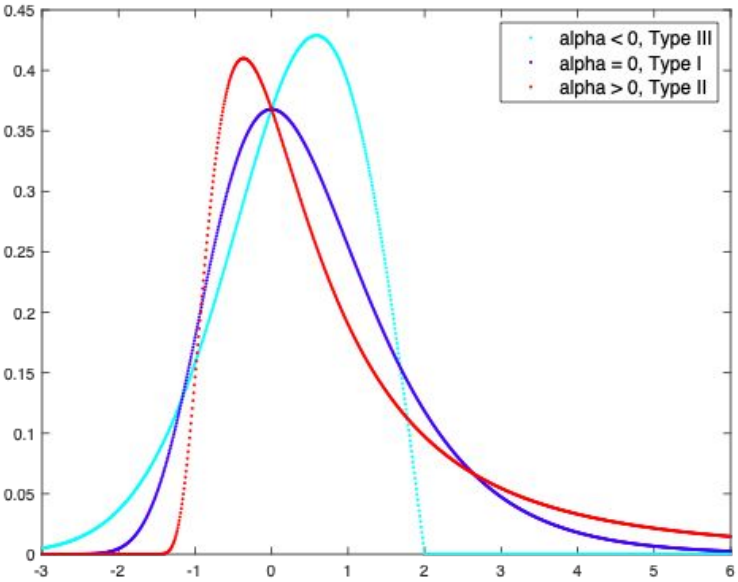


3. Estimate distributions of wind tunnel data



- Tested **normal, gamma, Gumbel, and general extreme value distributions** (GEV)
- Used Q-Q plot tests to determine which distribution fit best

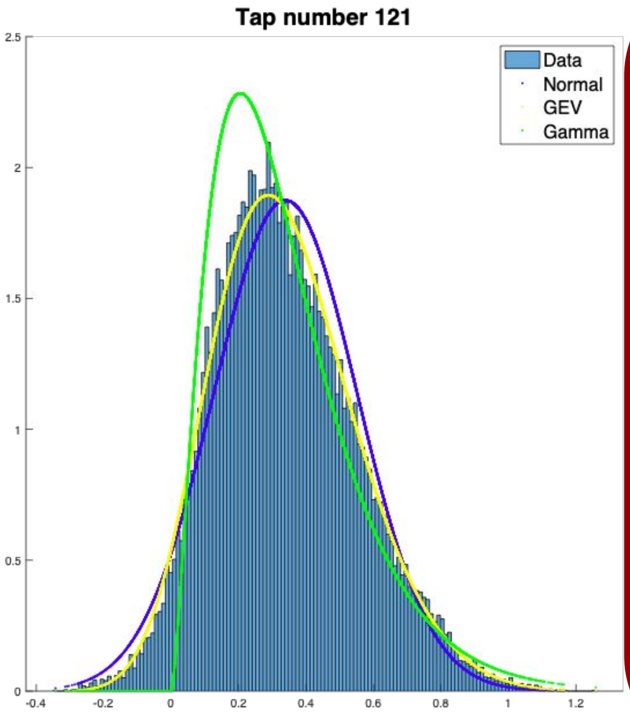
GEV distribution was added for its versatility

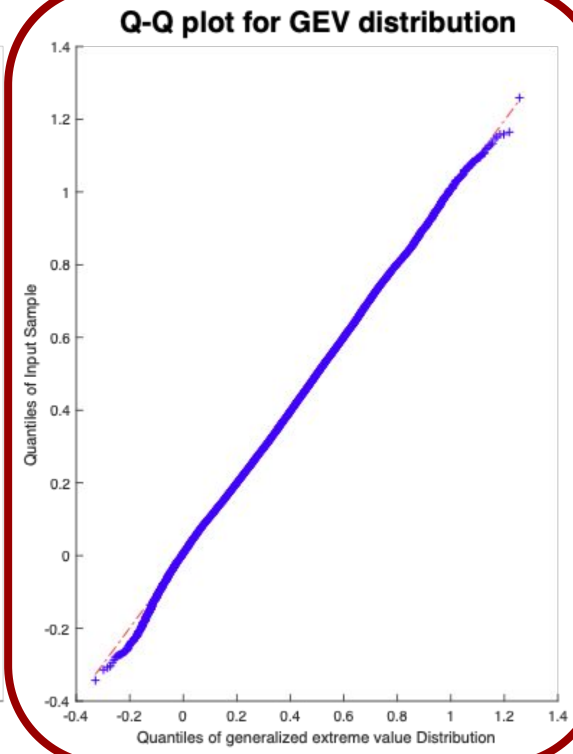


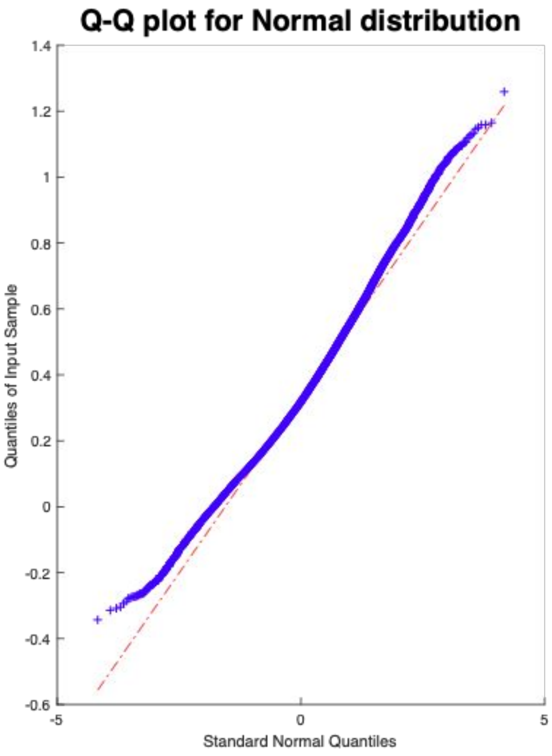
- μ, σ, α are location, scale and shape parameters
- α determines Type I (Gumbel), Type II (Frechet) or Type III (Weibull) distribution

$$f_{\mathrm{GEV}}(x;\mu,\sigma,\alpha) = \frac{1}{\sigma}\exp\left(-\left[1+\alpha\left(\frac{x-\mu}{\sigma}\right)\right]^{-1/\alpha}\right)\left[1+\alpha\left(\frac{x-\mu}{\sigma}\right)\right]^{-1/\alpha-1}$$

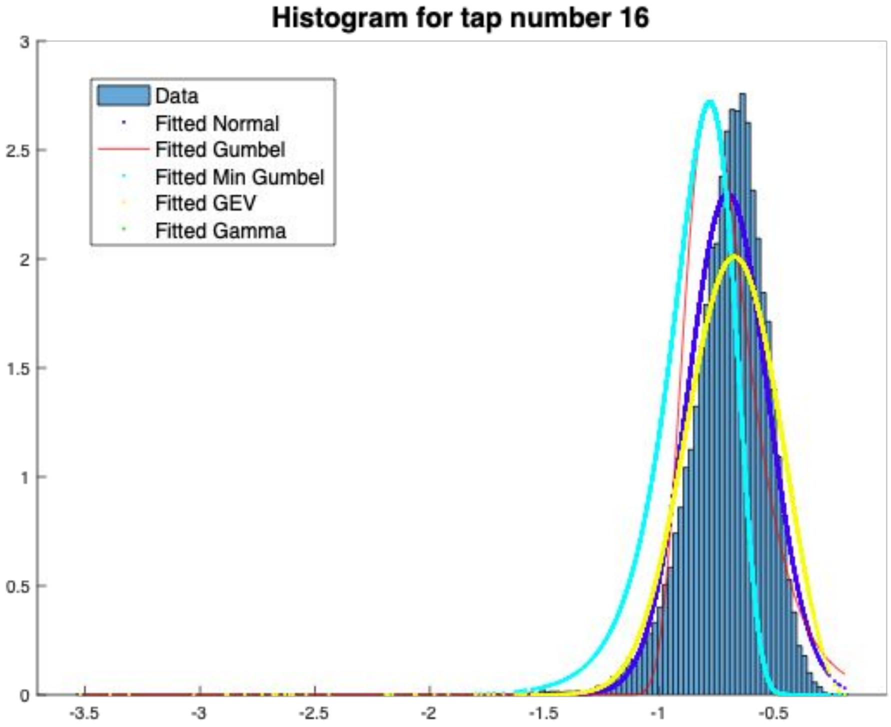
3. Estimate distributions of wind tunnel data

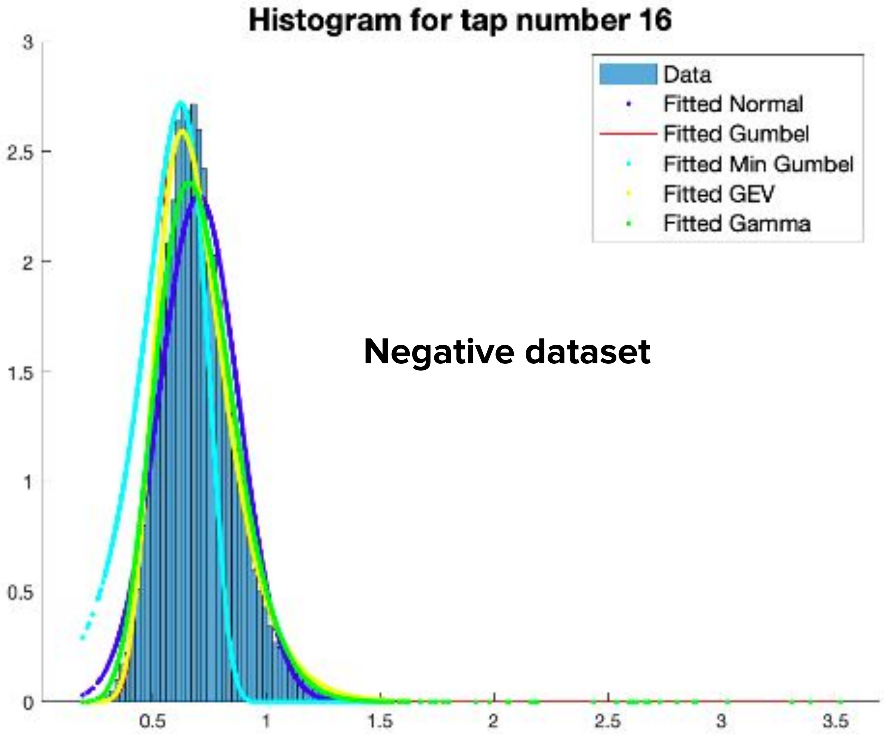






Negative values gave us some trouble until we flipped them





Negative dataset

4. Create a model for wind loads

4.1. Find correlation function $C(\zeta)$ of data

4.2. Provide values for ν_k

4.3. Estimate values for σ_k using $C(\zeta) = \sum^{m}_{k=1}\sigma_k^2 \cos(\nu_k\zeta)$

4.4 Create function $G(t)$ for model

4.5 Convert CDF of $G(t)$ to GEV distribution

4. Create a model for wind loads

4.0. Definitions and General Concepts

$$G(t) = \sum_{k=1}^{m} \sigma_k (A * cos(\nu_k * t) + B * sin(\nu_k * t))$$

$$A, B \sim N(0, 1)$$

$$E[G(t)] = E[\sum_{k=1}^{m} \sigma_k (A * cos(\nu_k * t) + B * sin(\nu_k * t))]$$

$$E[G(t)] = \sum_{k=1}^{m} \sigma_k * E[(A * cos(\nu_k * t) + B * sin(\nu_k * t))]$$

$$E[G(t)] = \sum_{k=1}^{m} \sigma_k * (E[A] * cos(\nu_k * t) + E[B] * sin(\nu_k * t))$$

$$E[G(t)] = 0$$

4. Create a model for wind loads

4.0. Definitions and General Concepts

$$Corr[G(t), G(s)] = E[G(t) * G(s)]$$

$$Corr[G(t), G(s)] = E[\sum_{k,j=1}^{m} \sigma_k * (A_k * cos(\nu_k * t) + B_k * sin(\nu_k * t)) * \sigma_j * (A_j * cos(\nu_j * s) + B_j * sin(\nu_j * s))]$$

$$Corr[G(t), G(s)] = E[\sum_{k,j=1}^{m} \sigma_k \sigma_j * (A_k * A_j * cos(\nu_k * t) * cos(\nu_j * s) + A_j * B_k * sin(\nu_k * t) * cos(\nu_j * s)...$$

$$A_k * B_j * cos(\nu_k * t) * sin(\nu_j * s) + B_k * B_j * sin(\nu_k * t) * sin(\nu_j * s))]$$

$$Corr[G(t), G(s)] = \sum_{k,j=1}^{m} \sigma_k \sigma_j * (E[A_k * A_j] * cos(\nu_k * t) * cos(\nu_j * s) + E[A_j * B_k] * sin(\nu_k * t) * cos(\nu_j * s)...$$

$$E[A_k * B_j] * cos(\nu_k * t) * sin(\nu_j * s) + E[B_k * B_j] * sin(\nu_k * t) * sin(\nu_j * s))$$

$$Corr[G(t), G(s)] = \sum_{k=1}^{m} \sigma_k^{2} * (cos(\nu_k * t) * cos(\nu_k * s) + sin(\nu_k * t) * sin(\nu_k * s))$$

$$\zeta = s - t$$

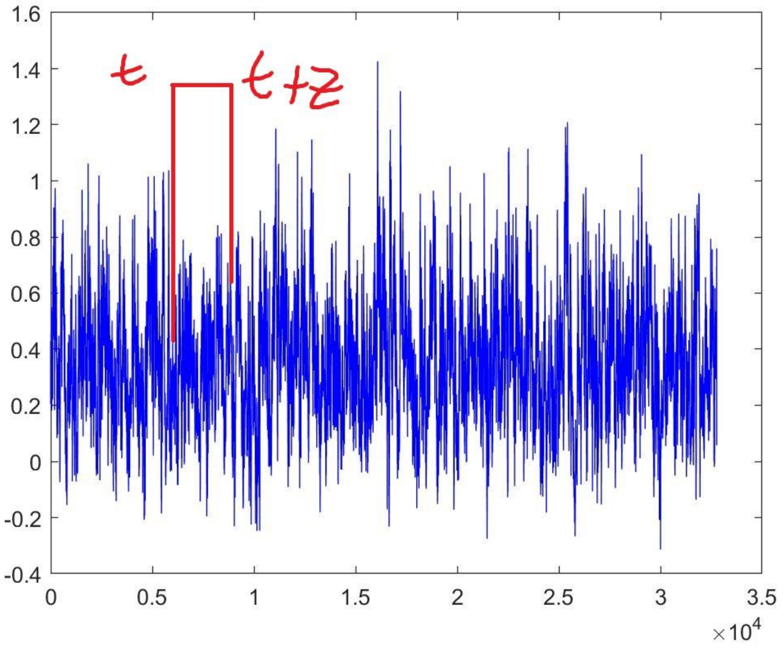
$$Corr[G(t), G(s)] = \sum_{k=1}^{m} \sigma_k^{2} * cos(\nu_k * \zeta)$$

$$\zeta = 0$$

$$Corr[G(t), G(s)] = E[G(t) * G(t)] = Var[G(t)] = \sum_{k=1}^{m} \sigma_k^{2}$$

4. Create a model for wind loads

4.1. Find correlation function C(ζ)

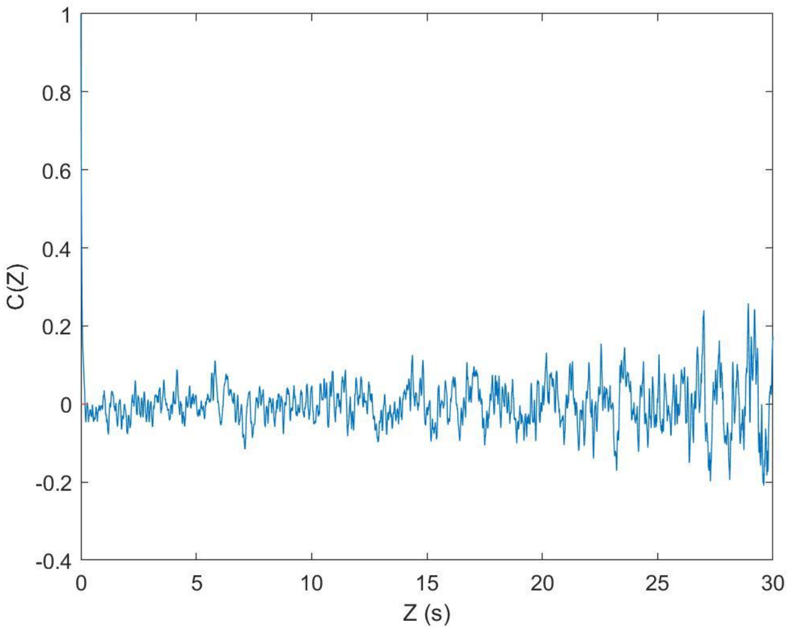


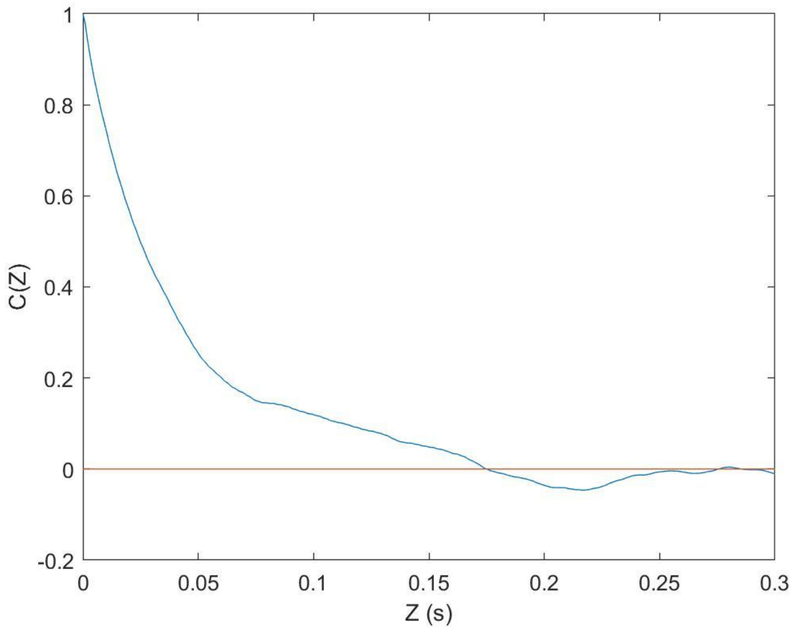
- Correlation as a function between two points in the data series for all data points

```
for k=1:kk

    C(k)=corr(dataset(1:n-Tstep*k,tap),dataset(Tstep*k+1:n,tap));

    if C(k)>0
        Count=Count+1*stopcount;
    else
        stopcount=0;
    end

end

C=[1,C];
```

4. Create a model for wind loads

4.1. Find correlation function C(ζ)





4. Create a model for wind loads

4.2. Provide values for v_k

$$v_1 = \frac{2 * \pi}{\tau}$$

$$v_k = v_1 * k$$

4. Create a model for wind loads

4.3. Estimate values for σ_k

$$C(\zeta) = \sum_{k=1}^{m} \sigma_k^2 cos(\nu_k * \zeta)$$

$$C(0) = 1$$

$$\sum_{k=1}^{m} \sigma_k^2 = 1$$

$$C(\zeta_1) = \sigma_1^2 cos(\nu_1 * \zeta_1) + \sigma_2^2 cos(\nu_2 * \zeta_1) + ...\sigma_m^2 cos(\nu_m * \zeta_1)$$

$$\overrightarrow{C(\zeta)} = \underline{A} * \overrightarrow{\sigma^2}$$

4. Create a model for wind loads

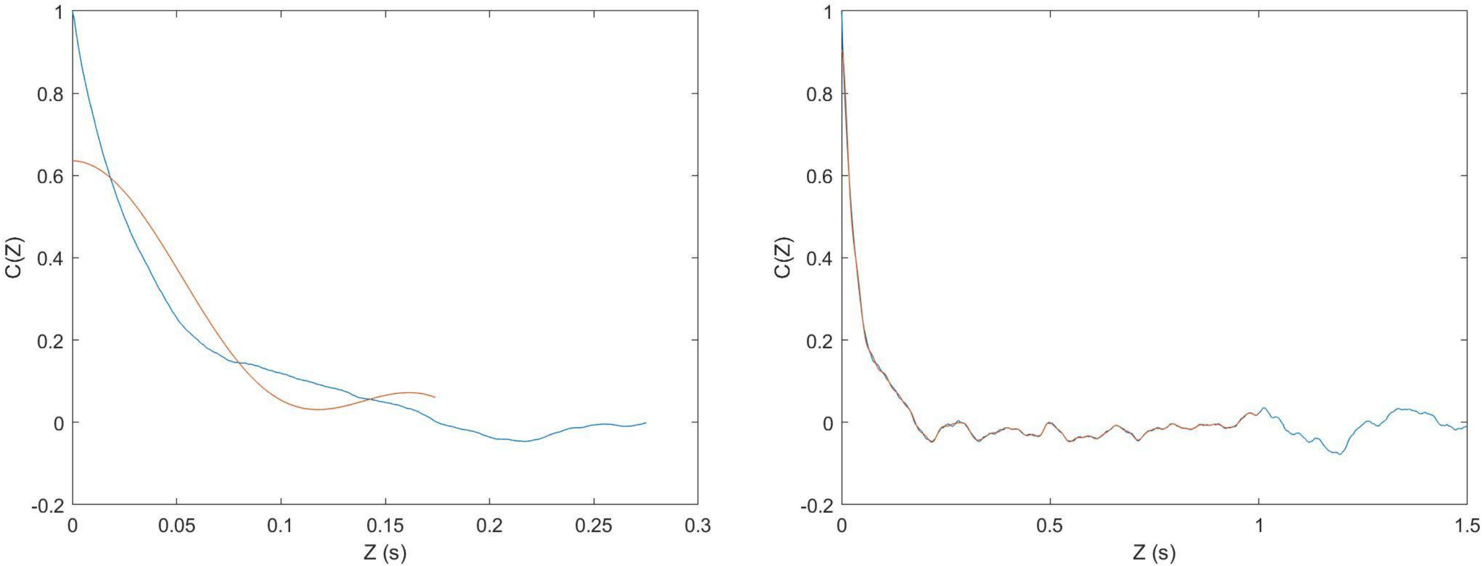
4.3. Estimate values for σ_k

- σ_k^2 cannot be < 0 because we need a real σ
- Solution was to use least squares non-negative regression (lsqnonneg)
- Uses iteration and Sum of Squared Errors (SSE) to estimate σ values and returns approximate values

4. Create a model for wind loads

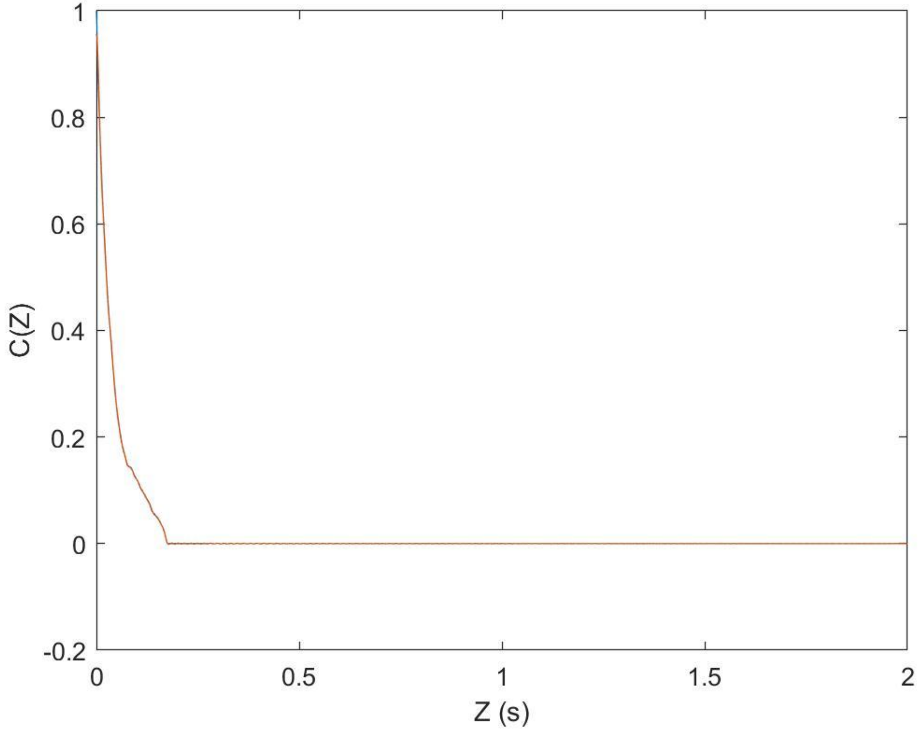
4.3. Estimate values for σ_k

- Accuracy of regression is dependent on the number of samples it is provided.



4. Create a model for wind loads

4.3. Estimate values for σ_k



4. Create a model for wind loads

4.4 Create function G(t) for model

$$G(t) = \sum_{k=1}^{m} \sigma_k (A * cos(\nu_k * t) + B * sin(\nu_k * t))$$

$$A, B \sim N(0,1)$$

```
t=0:0.001:T;
SIG=SIGS.^.5;
CT=cos(Vk'*t);
ST=sin(Vk'*t);
ns=1;
for i=1:ns
    G(i,:)=zeros(1,T*1000+1);
    for k=1:m
        A=randn(1,1);
        B=randn(1,1);
        G(i,:)=G(i,:)+SIG(k)*(A*CT(k,:)+B*ST(k,:));
    end
end
```

4. Create a model for wind loads

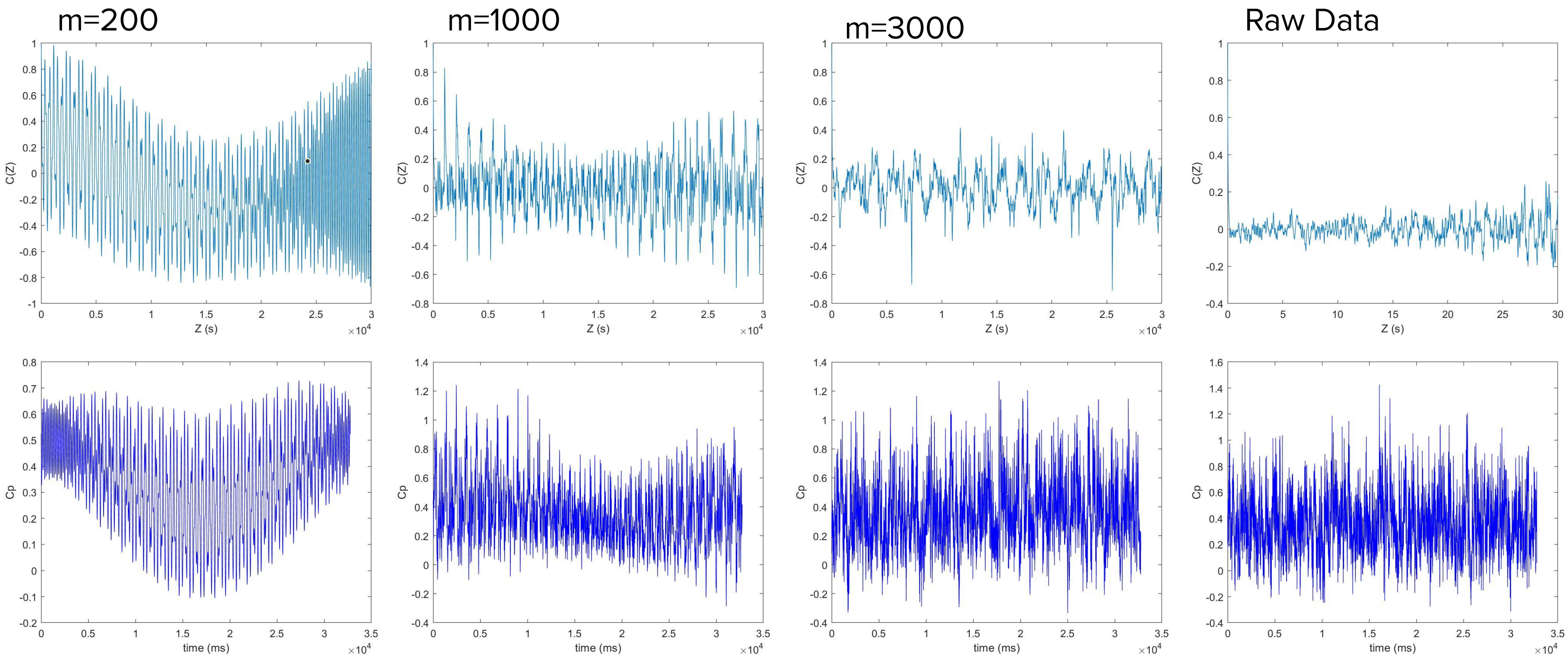
4.5 Convert CDF of G(t) to GEV distribution

$$X(t) = F^{-1}(\Phi(G(t)))$$

```
model=gevinv(normcdf(G,0,sum(SIGS)),p(1),p(2),p(3))';
```

- Location parameter, **μ = p(3)**
- Scale parameter, **σ = p(2)**
- Shape parameter, **α = p(1)**
- SIGS=$\sigma_k^{\ 2}$

Sanity Check



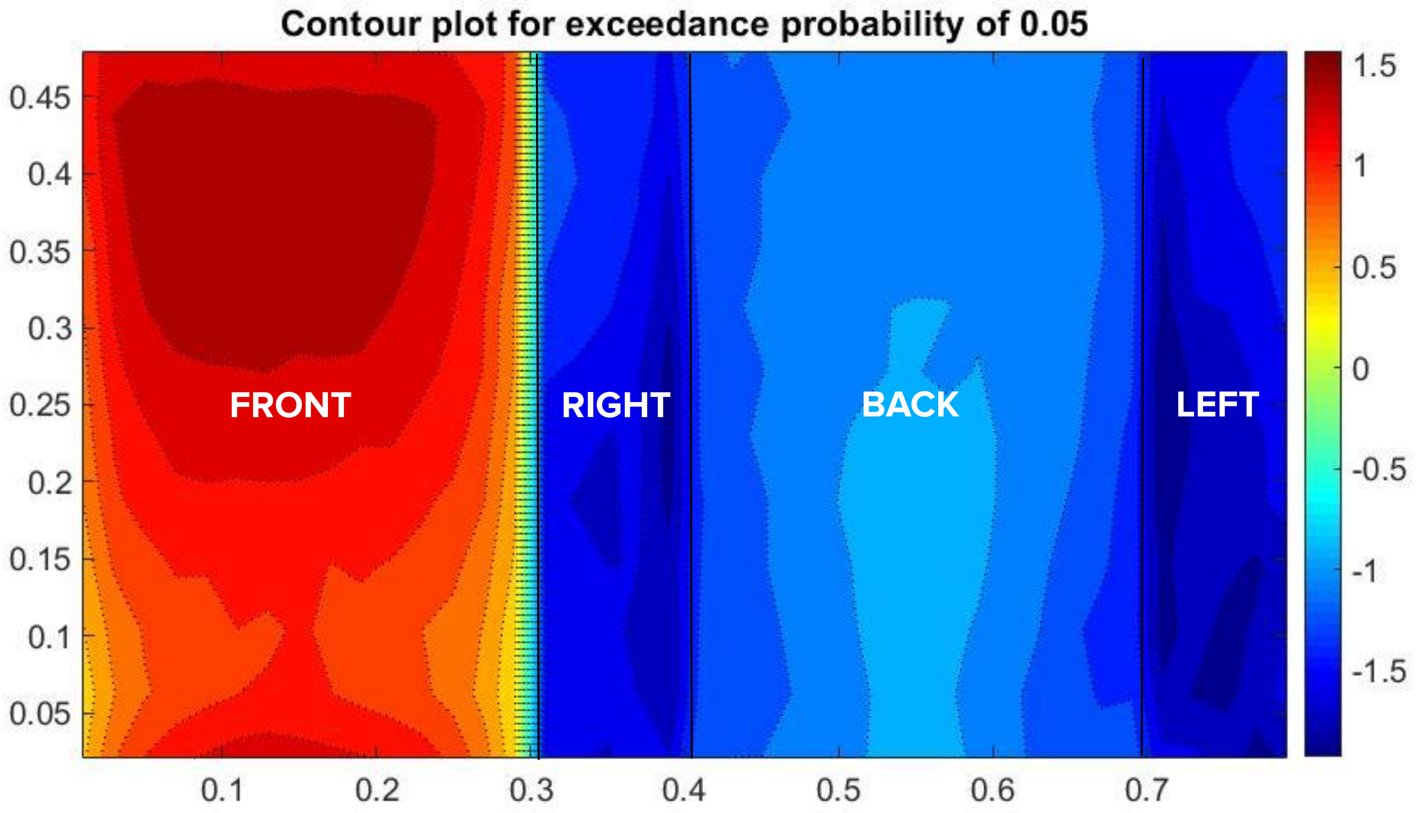
5. Generate wind load samples and find maximum

```
for k=1:ns
G(:,k)=sum(SIG.* (A(:,k).*CT+B(:,k).*ST));
end

model = zeros(32769,10);
Max = zeros(ns,1);

for k=1:ns
model(:,k)=gevinv(normcdf(G(:,k),0,sum(SIGS)),p(1),p(2),p(3))';
Max(k)=max(model(:,k));
end

[f,x]= ecdf(Max);
IND = sum(f<(1-EXC));
Values =x(IND);
Values=Values*Didwenegate;
end
```

6. Estimate pressure coefficients for exceedances

- m = 10, ns = 1000
- 32 minutes to run code!

7.1. Create contour plots



7.2. Draw conclusions

- Top half of front face has greatest wind load
- The sides have high negative pressure coefficients
- Need faster processors for more accurate results

Questions?

APPENDIX — Pressure Coefficients

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho_\infty V_\infty^2} = \frac{p - p_\infty}{p_0 - p_\infty}$$

where:

- p is the static pressure at the point at which pressure coefficient is being evaluated
- p_∞ is the static pressure in the freestream (i.e. remote from any disturbance)
- p_0 is the stagnation pressure in the freestream (i.e. remote from any disturbance)
- ρ_∞ is the freestream fluid density (Air at sea level and 15 °C is 1.225 $\mathrm{kg/m^3}$)
- V_∞ is the freestream velocity of the fluid, or the velocity of the body through the fluid