

# 1 Classical Ising Model

## 1.1 Structure of the code

The `ising` executable program calculates the expectation values of various observables and physical quantities derived from them, along with error bars. Each iteration of the program runs for a single  $(L, T)$  pair. The program is structured into four parts

**Part 1** A preliminary Monte Carlo run to collect autocorrelated observable measurements after the system equilibrates.

**Part 2** Calculation of the integrated autocorrelation time  $\tau$  using the autocorrelated measurements.

**Part 3** The main Monte Carlo run with observable measurements being taken after every  $2\tau$  MC sweeps to ensure collection of uncorrelated measurements.

**Part 4** Calculating observable expectation values and derived physical quantities with error bars using jackknife binning method.

After obtaining the results of expectation values and derived physical quantities (with error bars) for a range of temperatures, we perform **Finite Size Scaling** analysis on the data to extract the *critical exponents* of the 2D Ising Model.

## 1.2 Wolff Cluster algorithm

To perform the Monte Carlo sweeps, we use the Wolff Cluster updates which satisfies the required detailed balance condition. The algorithm that we use to implement this update scheme is outlined below, and is taken from [3].

**algorithm** wolff-cluster

**begin**

$i := \text{random particle};$

$\mathcal{C} := \{i\};$

$\mathcal{F}_{\text{old}} := \{i\};$

**while**  $\mathcal{F}_{\text{old}} \neq \{\}$  **do**

**begin**

$\mathcal{F}_{\text{new}} := \{\};$

**for**  $\forall i \in \mathcal{F}_{\text{old}}$  **do**

**begin**

**for**  $\forall j$  neighbor of  $i$  with  $S_i = S_j$ ,  $j \notin \mathcal{C}$  **do**

**begin**

**if**  $\text{ran}[0, 1] < p$  **then**

**begin**

$\mathcal{F}_{\text{new}} := \mathcal{F}_{\text{new}} \cup \{j\};$

$\mathcal{C} := \mathcal{C} \cup \{j\};$

```

        end
    end
end
 $\mathcal{F}_{\text{old}} := \mathcal{F}_{\text{new}};$ 
end
for  $\forall i \in \mathcal{C}$  do
     $S_i := -S_i;$ 
end
end
    
```

The above algorithm shows how a single cluster update/flip is implemented and is termed as one MC sweep.

### 1.3 Autocorrelation Time

Since Monte Carlo in general generates statistically correlated configurations (and measurements), we take measurements after every few time steps to get statistically uncorrelated measurements. The *autocorrelation function* gives us a measure of correlations between subsequent measurements. For an observable  $\mathcal{O}$ , the autocorrelation function is

$$A_{\mathcal{O}}(t) = \frac{1}{N-t} \sum_{k=0}^{N-t-1} \frac{(\mathcal{O}(k) - \langle \mathcal{O} \rangle) \cdot (\mathcal{O}(k+t) - \langle \mathcal{O} \rangle)}{\langle \mathcal{O}^2 \rangle - \langle \mathcal{O} \rangle^2} \quad (1.1)$$

where  $\mathcal{O}(k)$ 's are the measurements taken for each Monte Carlo step, and  $N$  is the total number of Monte Carlo sampling steps. Also, the averages in (1.1) are only taken over the first  $N - t$  measurements.

$$\langle \mathcal{O}^n \rangle = \frac{1}{N-t} \sum_{k=0}^{N-t-1} [\mathcal{O}(k)]^n$$

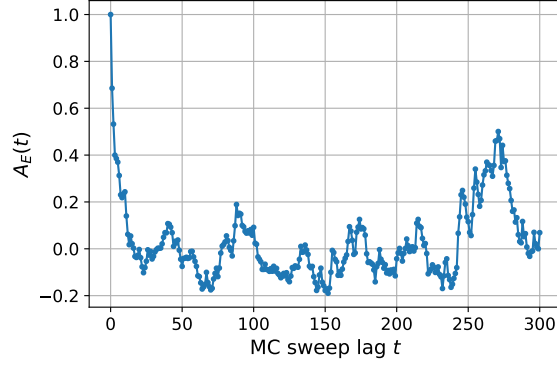
The autocorrelation function  $A_{\mathcal{O}}(t)$  is expected to fall off as an exponential

$$A_{\mathcal{O}}(t) \sim e^{-|t|/\tau}$$

where  $\tau$  is defined as the *autocorrelation time*. For an observable  $\mathcal{O}$ , we define the *integrated autocorrelation time*  $\tau_{\text{int}}$  as a discretized integration sum

$$\tau_{\text{int}, \mathcal{O}} = \frac{1}{2} \sum_{t=-\infty}^{\infty} A_{\mathcal{O}}(t) = \frac{1}{2} + \sum_{t=1}^{\infty} A_{\mathcal{O}}(t)$$

However, this natural estimator isn't *quite* correct because the autocorrelations  $A_{\mathcal{O}}(t)$  for  $t \gg \tau_{\text{int}}$  contain much noise but little signal. To fix this, we implement the *automatic*



**Figure 1:** Energy density autocorrelation function  $A_E(t)$  for  $L = 128$ ,  $T = 2.26$ .

*windowing algorithm* [5] by introducing a cutoff  $M$  on the sum

$$\tau_{\text{int},\mathcal{O}} = \frac{1}{2} + \sum_{t=1}^M A_{\mathcal{O}}(t) \quad (1.2)$$

$M$  is chosen as the smallest integer such that  $M \geq 6\tau_{\text{int},\mathcal{O}}(M)$ . The algorithm is summarized as follows

1. Start with some small value of  $M$ , say  $M = 50$ , and evaluate  $\tau_{\text{int},\mathcal{O}}(50)$ .
2. Check if  $50 \geq 6\tau_{\text{int},\mathcal{O}}(50)$ .
3. If not, increase the value of  $M$  and repeat until you find a  $M$  where  $M \geq 6\tau_{\text{int},\mathcal{O}}(M)$ .

The autocorrelation time  $\tau_{\text{int}}$  of a given  $(L, T)$  configuration is the maximum of autocorrelation times for all observables except magnetization  $m$  (due to lack of SSB in finite lattices).

$$\tau_{\text{int}} = \max \{ \tau_{\text{int},|m|}, \tau_{\text{int},m^2}, \tau_{\text{int},m^4}, \tau_{\text{int},E}, \tau_{\text{int},E^2} \} \quad (1.3)$$

After running the preliminary MC run (Part 1) with measurements taken after every MC step to calculate the integrated autocorrelation time  $\tau_{\text{int}}$ , we perform the main MC run (Part 3) with measurements taken after every  $2\tau_{\text{int}}$  steps.

## 1.4 Jackknife Analysis

We begin by binning the  $N$  measurements of observable  $\mathcal{O}$  into  $N_B$  non-overlapping bins of length  $k$  such that  $N_B k = N$ , and construct a shorter time series of bin averages. The bin average for the  $j^{\text{th}}$  bin is defined as

$$\mathcal{O}_j^{(B)} \equiv \frac{1}{k} \sum_{i=0}^{k-1} \mathcal{O}_{jk+i}$$

Below is a visual representation of the binning process

$$\begin{array}{c}
 \underbrace{\{\mathcal{O}_0, \mathcal{O}_1 \dots \mathcal{O}_{k-1}\}}_{\overline{\mathcal{O}}_0^{(B)}} \underbrace{\{\mathcal{O}_k, \mathcal{O}_{k+1} \dots \mathcal{O}_{2k-1}\}}_{\overline{\mathcal{O}}_1^{(B)}} \dots \dots \dots \underbrace{\{\mathcal{O}_{(N_B-1)k}, \mathcal{O}_{(N_B-1)k+1} \dots \mathcal{O}_{N-1}\}}_{\overline{\mathcal{O}}_{N_B-1}^{(B)}} \\
 \Downarrow \\
 \{\overline{\mathcal{O}}_0^{(B)}, \overline{\mathcal{O}}_1^{(B)}, \overline{\mathcal{O}}_2^{(B)} \dots, \overline{\mathcal{O}}_{N_B-1}^{(B)}\}
 \end{array}$$

Knowing the bin averages for the observables  $\mathcal{O} \in \{E, E^2, |m|, m^2, m^4\}$ , the bin estimates of the derived quantities are estimated as

$$\begin{aligned}
 \chi_j^{(B)} &= \beta L^2 \left( \overline{m^2}_j^{(B)} - \left[ \overline{|m|}_j^{(B)} \right]^2 \right) \\
 C_{Vj}^{(B)} &= \beta^2 L^2 \left( \overline{E^2}_j^{(B)} - \left[ \overline{E}_j^{(B)} \right]^2 \right) \\
 U_{Lj}^{(B)} &= 1 - \frac{\overline{m^4}_j^{(B)}}{3 \left[ \overline{m^2}_j^{(B)} \right]^2}
 \end{aligned}$$

We denote these derived quantities by  $\rho$  such that  $\rho \in \{\chi, C_V, U_L\}$ . The means over the bin averages (for  $\mathcal{O}$ ) and bin estimates (for  $\rho$ ) are also calculated

$$\overline{\mathcal{O}} = \overline{\overline{\mathcal{O}}_j^{(B)}} = \frac{1}{N_B} \sum_{i=0}^{N_B-1} \overline{\mathcal{O}}_i^{(B)} \quad (1.4a)$$

$$\overline{\rho} = \overline{\rho_j^{(B)}} = \frac{1}{N_B} \sum_{i=0}^{N_B-1} \rho_i^{(B)} \quad (1.4b)$$

For the Jackknife error analysis, we begin by constructing the same number ( $N_B$ ) of Jackknife bins containing all data but the  $j^{\text{th}}$  bin of the previously mentioned binning method. The Jackknife averages for these new bins are defined as

$$\begin{aligned}
 \overline{\mathcal{O}}_j^{(J)} &= \frac{N \overline{\mathcal{O}} - k \overline{\mathcal{O}}_j^{(B)}}{N - k} \\
 \overline{\rho}_j^{(J)} &= \frac{N \overline{\rho} - k \rho_j^{(B)}}{N - k}
 \end{aligned}$$

The error calculated using the Jackknife bins then comes out as

$$\delta \overline{\mathcal{O}} = \sqrt{\frac{N_B - 1}{N_B} \sum_{j=0}^{N_B-1} \left( \overline{\mathcal{O}}_j^{(J)} - \overline{\mathcal{O}} \right)^2} \quad (1.5a)$$

$$\delta \rho = \sqrt{\frac{N_B - 1}{N_B} \sum_{j=0}^{N_B-1} \left( \overline{\rho}_j^{(J)} - \overline{\rho} \right)^2} \quad (1.5b)$$

## 1.5 Simulation results

The following parameters were used in the 2D Ising model simulations for lattice sizes of  $L = 8, 16, 32, 64, 128$ .

### For preliminary Monte Carlo (Part 1)

no of equilibration sweeps = 1.0e3

no of sampling sweeps = 1.0e4

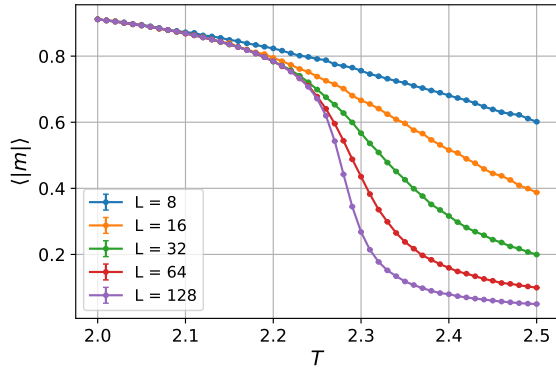
sampling step size = 1.0e0

### For the main Monte Carlo run (Part 3)

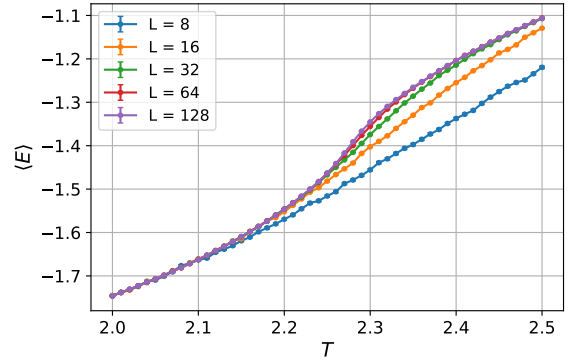
no of equilibration sweeps = 1.0e3

no of sampling sweeps =  $2\tau \cdot 2.0e4$

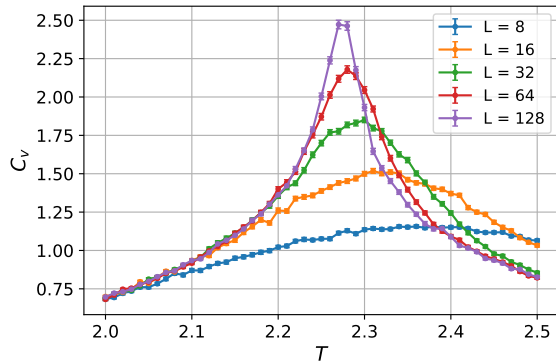
sampling step size =  $2\tau$



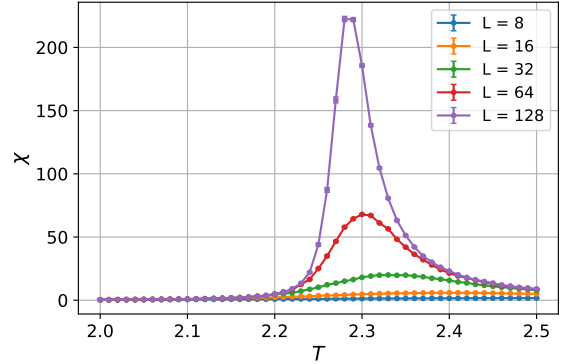
(a)  $|M|$  vs  $T$



(b)  $E$  vs  $T$

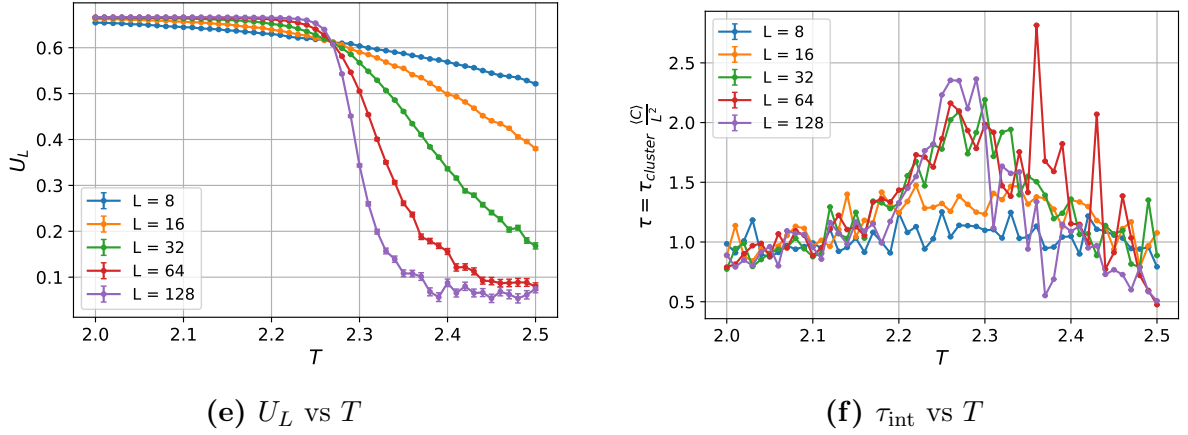


(c)  $C_V$  vs  $T$



(d)  $\chi$  vs  $T$

**Figure 2:** Variation of physical quantities with  $T$  for different lattice sizes  $L$ .



**Figure 2:** Variation of physical quantities with  $T$  for different lattice sizes  $L$ .

The Figure 2 above shows plots of expectation values of relevant observables, derived physical quantities and the autocorrelation times (scaled by  $\langle C \rangle / L^2$ ) as they vary with  $T$  for different lattice sizes  $L$ .

Furthermore, the critical temperature  $T_c$  is obtained by finding the point of intersection of different  $U_L$  vs  $T$  curves,  $\implies T_c \approx 2.269$ .

## 1.6 Finite-size scaling

According to the FSS hypothesis [4], observables and derived physical quantities close to  $T_c$  are a power of  $L$  multiplied by a non divergent function of  $tL^{1/\nu}$ , i.e.

$$Q(t, L) = L^{\zeta/\nu} g(tL^{1/\nu})$$

where  $t = (T - T_c)/T_c$ .

### FS relations for 2D Ising model

$$\begin{aligned} U_L &= g_U(tL^{1/\nu}) \\ |m| &= L^{-\beta/\nu} g_m(tL^{1/\nu}) \\ \chi &= L^{\gamma/\nu} g_\chi(tL^{1/\nu}) \\ C_V &= \ln(L) g_C(tL^{1/\nu}) \end{aligned}$$

We use the *data-collapse* method to estimate the critical exponents  $\beta$ ,  $\gamma$ , and  $\nu$  ( $\alpha = 0$  for 2D Ising model, so FSS hypothesis doesn't hold) by collapsing the curves corresponding to different lattice sizes  $L$  on a single unknown master curve  $g(x)$ .

We use the following cost function  $P_b$  to as a measure of the collapse, which is essentially a normalized sum of residues, which we get by slightly modifying the cost function

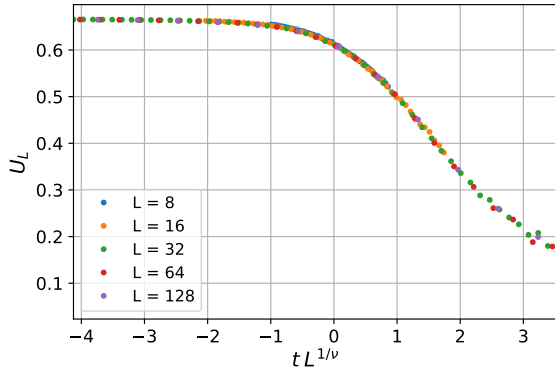
given in [1].

$$P_b = \frac{1}{N_{\text{over}}} \sum_p \sum_{j \neq p} \sum_i \frac{\left| L_j^{-\zeta/\nu} Q_{ij} - \varepsilon_p \left( L_j^{1/\nu} t_{ij} \right) \right|}{L_j^{-\zeta/\nu} Q_{ij} + \varepsilon_p \left( L_j^{1/\nu} t_{ij} \right)} \quad (1.6)$$

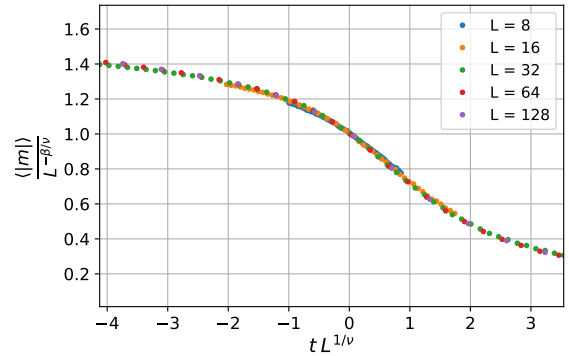
- $p$  indexes the data-set associated to length  $L_p$ .
- $j$  also indexes the data-set associated to length  $L_j$ , but doesn't include the set  $p$ .
- $i$  indexes the data-points inside the set associated to  $L_j$ .
- $N_{\text{over}}$  is the number of terms we sum over.

The critical exponents are obtained by minimizing the cost function  $P_b$  with respect to  $\beta, \gamma, \nu$ , i.e.

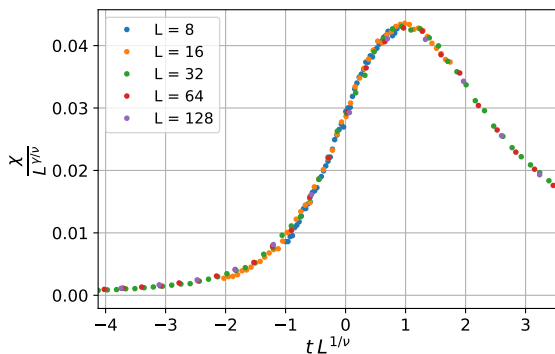
- $Q = U_L$ , minimize  $P_b(0, \nu)$  with respect to  $\nu$ .  $\longrightarrow \nu_0$
- $Q = |m|$ , minimize  $P_b(-\beta, \nu_0)$  with respect to  $\beta$ .  $\longrightarrow \beta_0$
- $Q = \chi$ , minimize  $P_b(\gamma, \nu_0)$  with respect to  $\gamma$ .  $\longrightarrow \gamma_0$



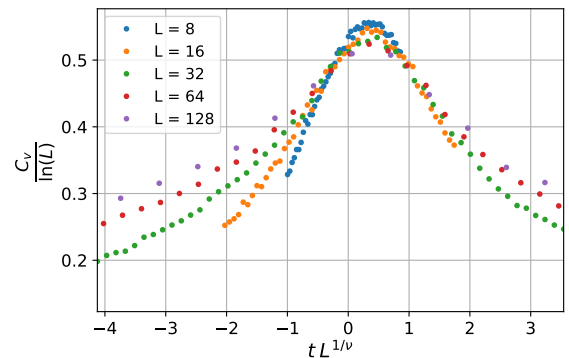
(a) Data collapse for  $U_L$



(b) Data collapse for  $|M|$



(c) Data collapse for  $\chi$



(d) Data collapse for  $C_V$

**Figure 3:** Data-collapse to extract the critical exponents.

The critical exponents estimated from the data collapse method are

$$\begin{aligned}\nu &= 0.9765 \\ \beta &= 0.1204 \\ \gamma &= 1.7301\end{aligned}$$

### 1.6.1 The quality of data collapse (theory)

The quality of the data collapse was defined as a reduced  $\chi^2$  statistic by [2], as follows

$$S = \frac{1}{\mathcal{N}} \sum_{i,j} \frac{(y_{ij} - Y_{ij})^2}{dy_{ij}^2 + dY_{ij}^2}, \quad (1.7)$$

where the values  $(y_{ij}, dy_{ij})$  are the scaled observations and its standard errors at  $x_{ij}$ , and the values  $(Y_{ij}, dY_{ij})$  are the estimated value of the master curve and its standard error at  $x_{ij}$ .

The quality  $S$  is the mean square of the weighted deviations from the master curve. As we expect the individual deviations  $y_{ij} - Y_{ij}$  to be of the order of the individual error  $\sqrt{dy_{ij}^2 + dY_{ij}^2}$  for an optimal fit, the quality  $S$  should attain its minimum  $S_{\min}$  at around 1 and be much larger otherwise.

Let  $i$  enumerate the system sizes  $L_i, i = 1, \dots, k$  and let  $j$  enumerate the parameters  $t_j, j = 1, \dots, n$  with  $t_1 < t_2 < \dots < t_n$ . The scaled data are

$$\begin{aligned}y_{ij} &:= L_i^{-\zeta/\nu} Q_{L_i, t_j} \\ dy_{ij} &:= L_i^{-\zeta/\nu} dQ_{L_i, t_j} \\ x_{ij} &:= L_i^{1/\nu} t_j\end{aligned}$$

The sum in the quality function  $S$  only involves terms for which the estimated value  $Y_{ij}$  of the master curve at  $x_{ij}$  is defined. The number of such terms is  $\mathcal{N}$ .

The master curve itself depends on the scaled data. For a given  $i, L_i$ , we estimate the master curve at  $x_{ij}$  by the two respective data from all the other system sizes which respectively enclose  $x_{ij}$ : for each  $i \neq i'$ , let  $j'$  be such that  $x_{i'j'} \leq x_{ij} \leq x_{i'(j'+1)}$ , and select the points  $(x_{i'j'}, y_{i'j'}, dy_{i'j'}), (x_{i'(j'+1)}, y_{i'(j'+1)}, dy_{i'(j'+1)})$ . Do not select points for some  $i'$ , if there is no such  $j'$ . If there is no such  $j'$  for all  $i'$ , the master curve remains undefined at  $x_{ij}$ .

Given the selected points  $(x_l, y_l, dy_l)$ , the local approximation of the master curve is the linear fit

$$y = mx + b$$



with weighted least squares. The weights  $w_l$  are the reciprocal variances,  $w_l := 1/dy_{ij}^2$ . The estimates and (co)variances of the slope  $m$  and intercept  $b$  are

$$\begin{aligned}\hat{b} &= \frac{1}{\Delta} (K_{xx}K_y - K_xK_{xy}) \\ \hat{m} &= \frac{1}{\Delta} (KK_{xy} - K_xK_y) \\ \hat{\sigma}_b^2 &= \frac{K_{xx}}{\Delta}, \hat{\sigma}_m^2 = \frac{K}{\Delta}, \hat{\sigma}_{bm} = -\frac{K_x}{\Delta}\end{aligned}$$

with  $K_{nm} := \sum w_l x_l^n y_l^m$ ,  $K := K_{00}$ ,  $K_x := K_{10}$ ,  $K_y := K_{01}$ ,  $K_{xx} := K_{20}$ ,  $K_{xy} := K_{11}$ ,  $\Delta := KK_{xx} - K_x^2$ . Hence, the estimated value of the master curve at  $x_{ij}$  is

$$Y_{ij} = \hat{m}x_{ij} + \hat{b}$$

with error propagation

$$dY_{ij}^2 = \hat{\sigma}^2 x_{ij}^2 + 2\hat{\sigma}_{bm}x_{ij} + \hat{\sigma}_b^2.$$

### 1.6.2 The quality of data collapse (results)

Using the algorithm in the previous subsection, the quality of the different data collapse fits are as follows

Quality of  $U_L$  vs  $tL^{1/\nu}$  collapse = 1.6933 (Fig. 3a)

Quality of  $\langle |m| \rangle L^{\beta/\nu}$  vs  $tL^{1/\nu}$  collapse = 22.4764 (Fig. 3b)

Quality of  $\chi L^{-\gamma/\nu}$  vs  $tL^{1/\nu}$  collapse = 0.7372 (Fig. 3c)

Quality of  $C_V/\ln(L)$  vs  $tL^{1/\nu}$  collapse = 6.9356 (Fig. 3d)

## 1.7 Problems that I'm facing

Following are some problems which I've not yet been able to resolve

1. **Interpreting the quality of data collapse as error** As discussed in our last meeting, you mentioned that the reduced  $\chi^2$  quality function  $S$  (Eq. (1.7)) itself is a measure of an error after looking at [2], so we can somehow normalize it to get the errors in physical quantities. However, even after reading the paper carefully, I still haven't understood how the authors calculated the errors, and to my knowledge, they haven't explicitly written down how they have calculated it.