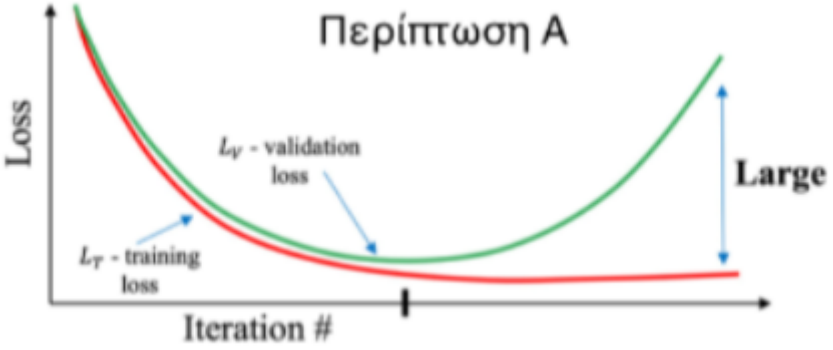
ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ ΚΑΙ ΒΑΘΙΑ ΜΑΘΗΣΗ
Ακαδημαϊκό έτος 2023-2024 Σειρά Αναλυτικών Ασκήσεων

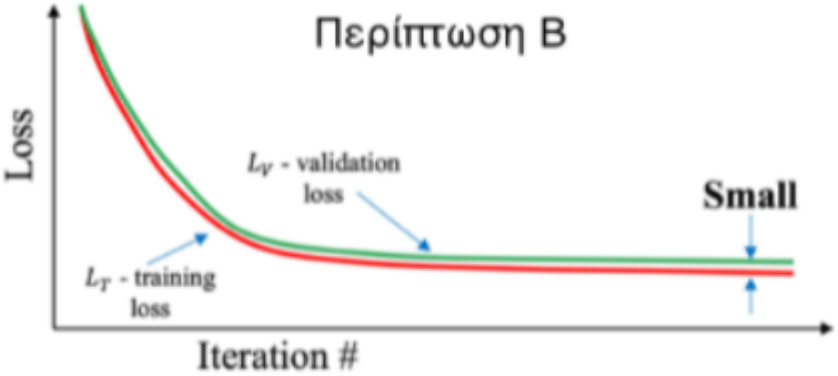
Θεοδώρα Εξάρχου
ΑΜ:03120865

Άσκηση 1 (Multi Layer Perceptron - Regularization)

1) Τι συμπέρασμα βγάζετε από το διάγραμμα για την αρχιτεκτονική του μοντέλου σας για καθεμία από τις περιπτώσεις Α και Β;



Περίπτωση Α:Αναφέρεται στο φαινόμενο της υπερπροσαρμογής (overfitting) σε μοντέλα μηχανικής μάθησης. Αν κατανοήσουμε σωστά την ανάλυση σου, η σημαντική και συνεχής μείωση του εκπαιδευτικού σφάλματος (Training loss) σε συνδυασμό με την αύξηση του σφάλματος επικύρωσης (Validation loss) υποδηλώνει ότι το μοντέλο έχει μάθει να προσαρμόζεται υπερβολικά στα δεδομένα εκπαίδευσης, αλλά δυσκολεύεται να γενικεύσει σε νέα δεδομένα.Αυτό συμβαίνει συνήθως όταν το μοντέλο έχει πολύ υψηλή χωρητικότητα, που σημαίνει ότι έχει την ικανότητα να μάθει τα πραγματικά αλλά και τα τυχαία μοτίβα που υπάρχουν στα δεδομένα εκπαίδευσης, συμπεριλαμβανομένου και του θορύβου. Ως αποτέλεσμα, το μοντέλο μπορεί να είναι πολύ ακριβές στο σύνολο δεδομένων εκπαίδευσης, αλλά να παρουσιάζει χαμηλή ακρίβεια σε νέα, μη εκπαιδευτικά δεδομένα



Περίπτωση Β:Η ανάλυση επικεντρώνεται στη συνεχή μείωση της συνάρτησης κόστους και στη σταθεροποίηση των τιμών της τόσο για το σύνολο εκπαίδευσης όσο και για το σύνολο επικύρωσης κατά τη διάρκεια των επαναλήψεων. Αυτό υποδεικνύει ότι το μοντέλο μαθαίνει αποτελεσματικά τα μοτίβα στα δεδομένα εκπαίδευσης χωρίς να προσθέτει περιττό θόρυβο.Η συνεχή μείωση της

συνάρτησης κόστους και η παρόμοια χαμηλή τιμή της για και τα δύο σύνολα υποδεικνύει ότι το μοντέλο δεν υπερπροσαρμόζεται στα δεδομένα εκπαίδευσης και έχει καταφέρει να επιτύχει μια ισορροπημένη γενίκευση. Το μικρό χάσμα μεταξύ της συνάρτησης κόστους για το σύνολο εκπαίδευσης και το σύνολο επικύρωσης υποδεικνύει ότι το μοντέλο είναι σε θέση να γενικεύει καλά σε νέα δεδομένα, δείχνοντας έτσι μια καλή ισορροπία μεταξύ εκπαίδευσης και γενίκευσης.

2) Ποια τιμή επαναλήψεων (εποχών) n είναι πιο πιθανό να οδηγεί στο καλύτερο δυνατό μοντέλο μάθησης

Περίπτωση Α:Η καλύτερη τιμή είναι κοντά στο μισό των εποχών που φαίνεται στο διάγραμμα, καθώς μετά έχουμε το overfitting.

Περίπτωση Β:Η καλύτερη τιμή είναι κοντά στο μισό των εποχών, καθώς το loss μειώνεται με πολύ αργό ρυθμό. Επομένως, μπορούμε να εξοικονομήσουμε πόρους με μικρή διαφορά στο loss.

3)Τεχνικές για την περίπτωση Α που θα μπορούσαν να βελτιώσουν την επίδοση του μοντέλου

- Εφαρμόζοντας Dropout rate στο μοντέλο
- Αλλάζοντας τις υπερπαραμέτρους

4)Εξηγήστε για ποιο λόγο είναι απαραίτητο το σύνολο ελέγχου (testing set) πέρα από τα σύνολα εκπαίδευσης και επικύρωσης.

Το σύνολο ελέγχου (testing set) είναι απαραίτητο πέρα από τα σύνολα εκπαίδευσης και επικύρωσης για την **Αντικειμενική Αξιολόγηση σε Άγνωστα Δεδομένα**.Το σύνολο ελέγχου βοηθά στον αντικειμενικό έλεγχο του μοντέλου σε άγνωστα δεδομένα, εξασφαλίζοντας ότι η απόδοση που παρατηρούμε είναι αντιπροσωπευτική της πραγματικής ικανότητας του μοντέλου να χειριστεί νέα δεδομένα που δεν έχουν χρησιμοποιηθεί στη διαδικασία εκπαίδευσης και επικύρωσης.

Άσκηση 2 (Representation Learning - Autoencoders)

(α) Ποια είναι η διάσταση των χαρακτηριστικών εισόδου xi στον auto-encoder?
Τα διανύσματα uou\mathbf{u}_{ou}uou και uuc\mathbf{u}_{uc}uuc αντιπροσωπεύουν τις λέξεις στο μοντέλο skipgram. Αυτά τα διανύσματα αποτελούν τις ενσωματώσεις (embeddings) των λέξεων και έχουν διάσταση 256. Όταν εκπαιδεύουμε έναν αυτοκωδικοποιητή, χρησιμοποιούμε αυτά τα διανύσματα ως είσοδο. Άρα, η διάσταση των χαρακτηριστικών εισόδου xi στον αυτοκωδικοποιητή θα είναι η ίδια με τη διάσταση των διανυσμάτων ενσωμάτωσης, δηλαδή 256.

(β) Ποια είναι η διάσταση των χαρακτηριστικών εξόδου yi του auto-encoder?
Συνεπώς, η διάσταση των χαρακτηριστικών εξόδου yi θα ταυτίζεται με τη διάσταση της εισόδου, δηλαδή 256.

(γ) Ποιά είναι η διάσταση της λανθάνουσας αναπαράστασης (latent representation) του auto-encoder?
Η διάσταση της λανθάνουσας αναπαράστασης (latent representation) ενός αυτοκωδικοποιητή είναι η διάσταση του μικρότερου κρυμμένου στρώματος (hidden layer) στον αυτοκωδικοποιητή. Εδώ, το μικρότερο κρυμμένο στρώμα έχει διάσταση 50. Επομένως, η διάσταση της λανθάνουσας αναπαράστασης του αυτοκωδικοποιητή είναι 50.

Άσκηση 3 (Recurrent Neural Networks) Χειρόγραφα Παρακάτω
Άσκηση 4 (Convolutional Neural Networks) Χειρόγραφα Παρακάτω

Άσκηση 5 (Generative models)

α)Variational Autoencoders
Χρησιμοποιούν κωδικοποιητή και αποκωδικοποιητή για την παραγωγή νέων δεδομένων. Εισάγουν κανονικοποίηση στον λανθάνοντα χώρο για να αποφεύγουν την υπερπροσαρμογή και να διασφαλίζουν τη συνεχή και πλήρη δομή του λανθάνοντα χώρου.Ο κωδικοποιητής συμπιέζει τα δεδομένα στον λανθάνοντα χώρο και ο αποκωδικοποιητής ανακατασκευάζει τα αρχικά δεδομένα από αυτόν τον χώρο.

β) Generative Adversarial Networks
Αποτελούνται από δύο νευρωνικά δίκτυα, μια γεννήτρια και έναν διαχωριστή, που ανταγωνίζονται μεταξύ τους. Η γεννήτρια παράγει ψευδή δεδομένα και ο διαχωριστής προσπαθεί να τα διακρίνει από τα πραγματικά.Η γεννήτρια εκπαιδεύεται για να παράγει δεδομένα που μπορούν να ξεγελάσουν τον διαχωριστή, ενώ ο διαχωριστής εκπαιδεύεται για να αναγνωρίζει τα ψευδή δεδομένα.

γ) Diffusion models
Βασίζονται σε διαδικασίες διάχυσης και αντιστρόφου διάχυσης θορύβου για τη δημιουργία νέων δειγμάτων. Προσθέτουν θόρυβο σταδιακά στα δεδομένα και μαθαίνουν να αντιστρέφουν αυτή τη διαδικασία.Θόρυβος προστίθεται σταδιακά στα δεδομένα σε μια προωθητική διαδικασία και αφαιρείται σε μια αντίστροφη διαδικασία διάχυσης.

Διαδικασία Εκπαίδευσης

Variational Autoencoders
Η εκπαίδευση περιλαμβάνει τη βελτιστοποίηση του κωδικοποιητή και του αποκωδικοποιητή για την ελαχιστοποίηση του σφάλματος ανακατασκευής.Εισάγεται κανονικοποίηση στον λανθάνοντα χώρο για να διατηρείται η δομή και να αποφεύγεται η υπερπροσαρμογή.

Generative Adversarial Networks
Εκπαιδεύουν τη γεννήτρια και τον διαχωριστή σε εναλλασσόμενες περιόδους εκπαίδευσης.Ο διαχωριστής εκπαιδεύεται για να διακρίνει πραγματικά από ψευδή δεδομένα, ενώ η γεννήτρια εκπαιδεύεται για να ξεγελάσει τον διαχωριστή.Η σύγκλιση είναι δύσκολη και απαιτεί προσεκτική ρύθμιση των υπερπαραμέτρων.

Diffusion models
Απαιτούν μεγάλο αριθμό βημάτων εκπαίδευσης (συνήθως μερικές χιλιάδες) για να παράγουν δείγματα υψηλής ποιότητας.Χρησιμοποιούν δυναμικές Langevin και προσδιοριστικές διαφορικές εξισώσεις για τη δημιουργία δειγμάτων.

Αποτελεσματικότητα και Απαιτήσεις

Οι **Variational Autoencoders (VAEs)** απαιτούν λιγότερους πόρους για εκπαίδευση σε σχέση με τα GANs. Μπορεί να παράγουν δείγματα χαμηλότερης ποιότητας σε σύγκριση με τα GANs. Τα **Generative Adversarial Networks (GANs)** απαιτούν περισσότερη μνήμη και χρόνο εκπαίδευσης. Η σύγκλιση είναι δύσκολη και συχνά ασταθής. Μπορούν να παράγουν εξαιρετικά ρεαλιστικά δείγματα όταν εκπαιδευτούν σωστά. Τα **Diffusion Models** παράγουν δείγματα υψηλής ποιότητας. Ωστόσο, η διαδικασία δειγματοληψίας είναι αργή και απαιτεί πολλές αξιολογήσεις του νευρωνικού δικτύου, καθιστώντας τα δύσκολα για εφαρμογές πραγματικού χρόνου.

Ομοιότητες και Διαφορές

Όλα χρησιμοποιούν νευρωνικά δίκτυα για τη δημιουργία δεδομένων και επιδιώκουν την εκμάθηση κατανομών δεδομένων για τη δημιουργία ρεαλιστικών δειγμάτων.Όσον αφορά τις διαφορες , Variational Autoencoders χρησιμοποιούν κωδικοποίηση και αποκωδικοποίηση με κανονικοποίηση του λανθάνοντα χώρου.Generative Adversarial Networks βασίζονται σε έναν ανταγωνισμό μεταξύ γεννήτριας και διαχωριστή.Diffusion Models προσθέτουν και αφαιρούν θόρυβο στα δεδομένα σε μια διαδικασία διάχυσης.

Πλεονεκτήματα και Μειονεκτήματα
Τα **Variational Autoencoders (VAEs)** έχουν ως πλεονεκτήματα την εύκολη εκπαίδευση και την καλή κανονικοποίηση του λανθάνοντα χώρου. Ωστόσο, μπορεί να παράγουν δείγματα χαμηλότερης ποιότητας. Τα **Generative Adversarial Networks (GANs)** έχουν την ικανότητα δημιουργίας ρεαλιστικών δειγμάτων, αλλά η εκπαίδευσή τους είναι δύσκολη, απαιτεί μεγάλη υπολογιστική ισχύ και προσεκτική ρύθμιση των υπερπαραμέτρων. Τα **Diffusion Models** παράγουν δείγματα υψηλής ποιότητας και έχουν καλή θεωρητική βάση, αλλά η διαδικασία δειγματοληψίας είναι αργή και η εφαρμογή τους σε πραγματικά προβλήματα είναι δύσκολη λόγω του υψηλού υπολογιστικού κόστους.

Άσκηση 6 (Graph Neural Networks) Χειρόγραφα Παρακάτω

Lesson 3

a) $c = \frac{1}{4}([4,0]^T + [0,4]^T + [0,0]^T + [0,0]^T = [1,1]^T$

$\Rightarrow c_1 = [1,1]^T$

$x_1 = \text{concat}(\langle start \rangle, c) = [0,0,0,1,1]^T$

$h_1 = g(W_{hh} \cdot h_0 + W_{hx} \cdot x_1) =$

$= g\left(\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot [0,0]^T + \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix} \cdot [0,0,0,1,1]^T\right) =$

$= g\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix}\right) \xrightarrow{ReLU} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$y_1 = W_{gh} \cdot h_1 = \begin{bmatrix} -5 & -5 \\ 0 & 3 \\ 1 & 2 \\ 2 & 2 \\ 3 & -1 \\ 2,9 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -10 \\ 3 \\ 3 \\ 4 \\ 2 \\ 2,9 \end{bmatrix}$ H cat

$y_1 = y_{cat}$

- $x_2 = \text{concat}(y_1, c_2) = [1, -2, 0, 1, 1]^T$

$h_2 = g(W_{hh} \cdot h_1 + W_{hx} x_2) =$

$= g\left(\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot [1\ 1]^T + \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix} \cdot [1, -2, 0, 1, 1]^T\right)$

$= g\left(\begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 2 \\ -1 \end{bmatrix}\right) = g\left(\begin{bmatrix} 3 \\ 0 \end{bmatrix}\right) = \begin{bmatrix} 3 \\ 0 \end{bmatrix}$

$$y_2 = W_{gh} \cdot h_2 = \begin{bmatrix} -5 & -5 \\ 0 & 3 \\ 1 & 2 \\ 2 & 2 \\ 3 & -1 \\ 2.9 & 0 \end{bmatrix} \begin{bmatrix} 3 \\ 0 \end{bmatrix} = \begin{bmatrix} -15 \\ 0 \\ 3 \\ 6 \\ 9 \\ 8.7 \end{bmatrix}$$ # staving

$y_2 = y_{staving}$

- $x_3 = \text{concat}(y_2, c_3) = [0, -1, -1, 1, 1]^T$

$$h_3 = g(W_{hh} \cdot h_2 + W_{hx} \cdot x_3) =$$

$$= g\left(\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 3 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \cdot [0, -1, -1, 1, 1]\right)$$

$$= g\left(\begin{bmatrix} 0 \\ 3 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix}\right) = g\left(\begin{bmatrix} 0 \\ 2 \end{bmatrix}\right) = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$$

$$y_3 = W_{gh} h_3 = \begin{bmatrix} -5 & -5 \\ 0 & 3 \\ 1 & 2 \\ 2 & 2 \\ 3 & -1 \\ 2.9 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 2 \end{bmatrix} = \begin{bmatrix} -10 \\ 6 \\ 4 \\ 4 \\ -2 \\ 0 \end{bmatrix}$$ # <stop>

Επομένως έχουμε <start> cat staving <stop>

6) Ο αυτοπροσοχικός RNN έχει μηχανισμό scaled dot-product για κάθε βήμα το διάνυσμα ερωτήματος είναι h_{t-1} και οι αναπαραστάσεις των χαρακτηριστικών εικόνων i_1, i_2, i_3, i_4 χρησιμοποιούνται ως κλειδιά (key) και ως τιμές (value)

$h_0 = 0$

Για το πρώτο χρονικό βήμα ($t=1$) υπολογίζετε τις πιθανοτικές προσοχής (attention probabilities) και το διάνυσμα εικόνας που προκύπτει α.

$$key = W^k x_i$$

$$value = W^v x_i$$

$$query = W^q x_i$$

↳ embedding matrix

$$Z = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

→ διάσταση διανύσματος

Attention Scores

$$score = QK^T = h_0 [i_1, i_2, i_3, i_4]^T = \begin{bmatrix} 0 \\ 0 \end{bmatrix}^T \begin{bmatrix} 4 & 0 \\ 0 & 4 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$= [0\ 0\ 0\ 0]$$

$$\text{Scaled score} = \frac{score}{\sqrt{d_k}} = \frac{[0\ 0\ 0\ 0)}{\sqrt{2}}$$

$P = softmax(Scaled\ Score)$

$$P = \frac{\exp(0)}{\sum_{k=1}^{4} \exp(0)} = 0,25$$

Για το υπολογισμό διανύσματος G

$$G = \sum_{j=i}^{4} P \cdot i_j = 0,25\,[4,0]^T + 0,25\,[0,4]^T + 0,25\,[0,0]^T + 0,25\,[0,0]^T$$

$$G = 0,25\,[4,0]^T + 0,25\,[0,4]^T = [1,0]^T + [0,1]^T = [1,1]^T$$

Το G είναι ομοιο με του πρώτου εφαπτόμενου δεν χρειάζεται να συνεχίσουμε τον έλεγχο θα έχουμε τον ίδιο λέξη **cat**

Άσκηση 4

AlexNet

Διαστάσεις εικόνων $227 \times 227 \times 3$ (έγχρωμες με RGB channels)

Φίλτρο $11 \times 11 \times 3$ στο πρώτο convolution layer.

Έχουν συνολικά 96 φίλτρα, stride ίσο με 4 και padding μηδενικά

α) Διαστάσεις στην έξοδο πρώτου convolutional layer.

$$W_2 = \frac{W_1 - F_w + 2P}{S_w} + 1 \xrightarrow{P=0} W_2 = \frac{W_1 - F}{S} + 1$$

$$H_2 = \frac{H_1 - F_h + 2P}{S_h} + 1 \xrightarrow{P=0} H_2 = \frac{H_1 - F}{S} + 1$$

$$P_2 = kP_1$$

$$W_2 = \frac{227 - 11}{4} + 1$$

$$H_2 = \frac{227 - 11}{4} + 1$$

$P_2 = 1$ για πρώτο φίλτρο

$W_2 = 55$

$H_2 = 55$

$P_2 = 1$

διαστάσεις $55 \times 55 \times 96$

και για $k = 96$ φίλτρα έχουμε $P_2 = 96 \cdot 1 = 96$

β) Αριθμό units στο πρώτο convolutional layer

$5 \cdot 5 \cdot 55 = 3025$ units σε ένα filter

Για 96 έχουμε 290400 units

γ) Επαναλαμβανόμενη παραμέτρων του 1ο conv layer με διαφορετικά βάρη

$[\text{width} \times \text{height} \times \text{channels} + 1] \times \text{filters} =$

$(11 \cdot 11 \cdot 3 + 1) \cdot 96 = 34.944$

δ) Feedforward layer. 256 units

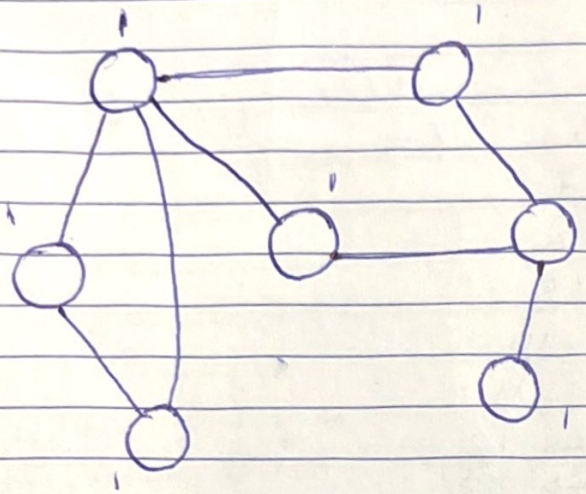
Numb. of par. = (Num. Input + Units) + Units

Numbers of parameters = $(227 \cdot 227 \cdot 3 \cdot 256) + 256$

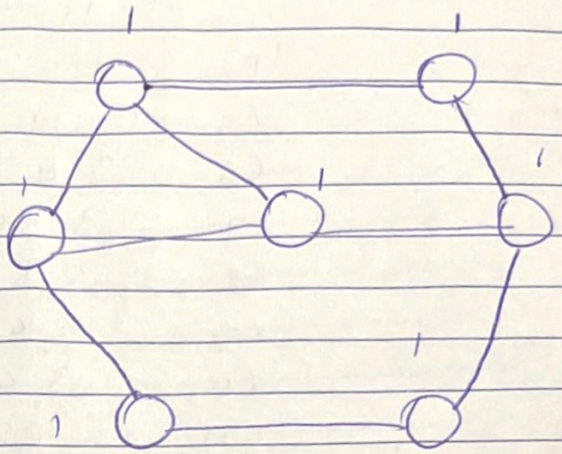
$= 39574528$

Lemon G

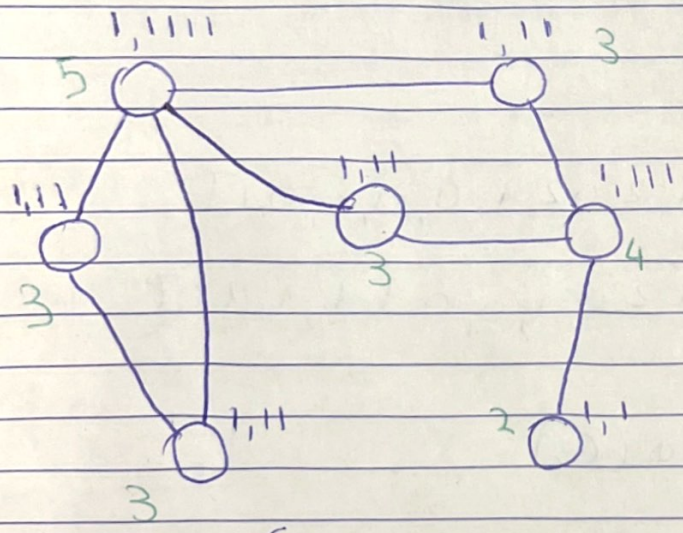
G_1



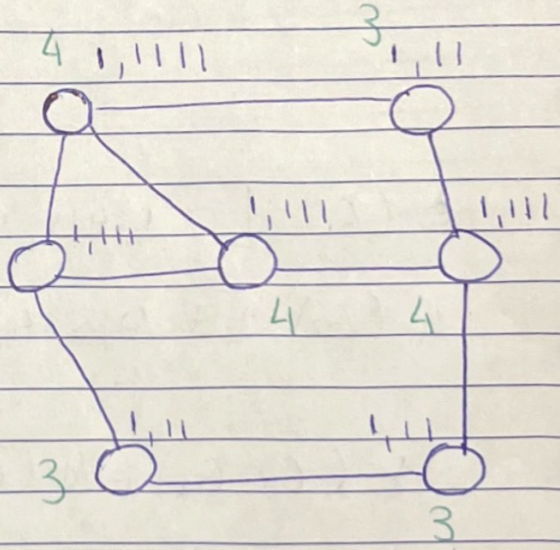
G_2



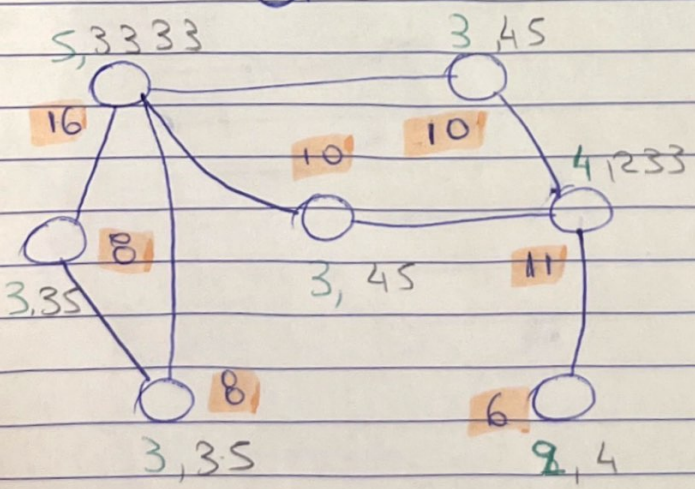
G_1



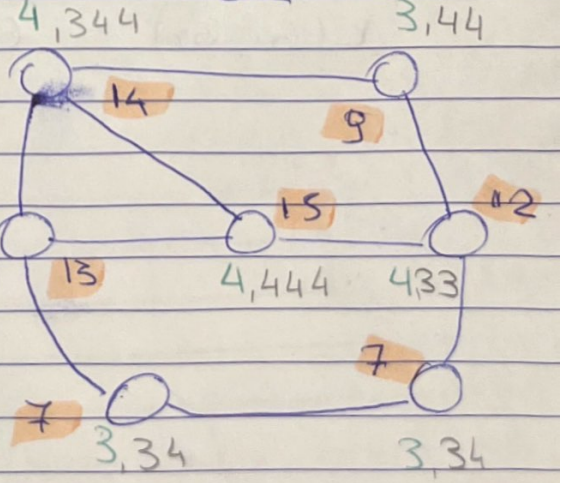
G_2



G_1



G_2



hash

C_2 1,1 → 2
C_3 1,11 → 3
C_4 1,111 → 4
C_5 1,1111 → 5
C_6 2,4 → 6
C_7 3,34 → 7
C_8 3,35 → 8
C_9 3,44 → 9
C_{10} 3,45 → 10
C_{11} 4,233 → 11
C_{12} 4,33 → 12
C_{13} 4,34 → 13
C_{14} 4,344 → 14
C_{15} 4,444 → 15
C_{16} 5,333 → 16

$\phi(G_1) = [7,1,4,1,1,1,0,2,0,2,1,0,0,0,0,1]$

$\phi(G_2) = [7,0,3,4,0,0,2,0,1,0,0,1,1,1,1,0]$

$K(G_1, G_2) = \phi(G_1)^T \phi(G_2) =$

$K(G_1, G_2) = 49 + 0 + 12 + 4 + 0$

$K(G_1, G_2) = 65$