-Revision Materials For a PDE Derivation (Proof)
Proof Dernation
1 Dernation
Starting from the fully Nom- linear poles taking
the form
(2, U+M.DxU+/Tx (OOTD2U) = F(,,,u,DxU,D2U)
(U(T.) = 9 detried on [O,T] XR
(u(T.)=g detined on [O,T] x Rd — O )
March 187 2 Controller
- let's define the diffusion process X in the domain
-let's define the diffusion process X in the domain  Parassocrated with Non-linear pole O given as
1 (300)
Xt=Xo+ (u(s, Xs)ds+ (J(s, Xs)dNs, text
L. T. T
Proof Dernation
Proof Dervation  Starting from the fully Nom- rinear poles taking
Starting from the fully Nome rinear poles taking the form
Starting from the fully Nome rinear poles taking the form
Starting from the fully Nome rinear poles taking the form  (20 + M.Dau + / Tr (OF Dau) = F(.,., U, Dau, Dau)
Starting from the fully Nome rinear poles taking the form  (20 + M.Dau + / Tr (OF Dau) = F(.,., U, Dau, Dau)
Starting from the fully Nome rinear poles taking the form
Starting from the fully Nom-Inner poles taking the form  [ ] Lu + M. D. Lu + 1/2 Tr (OF D. Lu) = F(., ., u, D. Lu, D. Lu)  [ u(T.) = g defined on [0,T] x Rd - 0  on Rd page 2
Starting from the fully Nom-Inner poles taking the form  [ ] Lu + M. D. Lu + 1/2 Tr (OF D. Lu) = F(., ., u, D. Lu, D. Lu)  [ u(T.) = g defined on [0,T] x Rd - 0  on Rd page 2
Starting from the fully Nom-Inner poles taking the form  [ ] Lu + M. D. Lu + 1/2 Tr (OF D. Lu) = F(., ., u, D. Lu, D. Lu)  [ u(T.) = g defined on [0,T] x Rd - 0  on Rd page 2
Starting from the fully Nom-Inner poles taking the form  [ ] Lu + M. D. Lu + 1/2 Tr (OF D. Lu) = F(., ., u, D. Lu, D. Lu)  [ u(T.) = g defined on [0,T] x Rd - 0  on Rd page 2
Starting from the fully Nom-Inner poles taking the form  [ ] Lu + M. D. Lu + 1/2 Tr (OF D. Lu) = F(., ., u, D. Lu, D. Lu)  [ u(T.) = g defined on [0,T] x Rd - 0  on Rd page 2
Starting from the fully Nom-Inner poles taking the form  [ ] Lu + M. D. Lu + 1/2 Tr (OF D. Lu) = F(., ., u, D. Lu, D. Lu)  [ u(T.) = g defined on [0,T] x Rd - 0  on Rd page 2
Starting from the fully Nom- linear poles taking  the form  (2eu + M. D. 2u + 1/2 Tr (00 T D. 2u) = F(., ., u, D. 2u, D. 2u)  (u(T.) = g detined on [0, T] x Rd - 0  on Ho page 2  - let's detine the diffusion process X in the domain  Rd associated with Non-linear pole (0 given as  t  Xt = Xo + Ju(s, Xs) ds + Ju(s, Xs) dNs, text
Starting from the fully Nom- linear poles taking  the form  (2eu + M. D. 2u + 1/2 Tr (00 T D. 2u) = F(., ., u, D. 2u, D. 2u)  (u(T.) = g detined on [0, T] x Rd - 0  on Ho page 2  - let's detine the diffusion process X in the domain  Rd associated with Non-linear pole (0 given as  t  Xt = Xo + Ju(s, Xs) ds + Ju(s, Xs) dNs, text
Starting from the fully Nom- linear poles taking  the form  (2eu + M. D. 2u + 1/2 Tr (00 T D. 2u) = F(., ., u, D. 2u, D. 2u)  (u(T.) = g detined on [0, T] x Rd - 0  on Ho page 2  - let's detine the diffusion process X in the domain  Rd associated with Non-linear pole (0 given as  t  Xt = Xo + Ju(s, Xs) ds + Ju(s, Xs) dNs, text
Starting from the fully Nom- linear poles taking  the form  (2eu + M. D. 2u + 1/2 Tr (00 T D. 2u) = F(., ., u, D. 2u, D. 2u)  (u(T.) = g detined on [0, T] x Rd - 0  on Ho page 2  - let's detine the diffusion process X in the domain  Rd associated with Non-linear pole (0 given as  t  Xt = Xo + Ju(s, Xs) ds + Ju(s, Xs) dNs, text
Starting from the fully Nom-linear poles taking the form  (244 + 11.0 x 4 + 1/2 Tr (ToT 0 x 4) = F(., ., 4, 0 x 4, 0 x 4)  (4(T.) = g dotined on [0, 7] x Rd — (0)  on 180 page 2  - let i de time the diffusion process X in the domain  18d associated with Non-linear pole (0) given as  t Xt = Xo + fu (s, Xs) ds + fr (s, Xs) dhs, tel  Xt = Xo + fu (s, Xs) ds + fr (s, Xs) dhs, tel  The general probabilistic representation of Solution to the  above pole (0) is thus given as
Starting from the fully Nom-linear poles taking the form  (244 + 11.0 x 4 + 1/2 Tr (ToT 0 x 4) = F(., ., 4, 0 x 4, 0 x 4)  (4(T.) = g dotined on [0, 7] x Rd — (0)  on 180 page 2  - let i de time the diffusion process X in the domain  18d associated with Non-linear pole (0) given as  t Xt = Xo + fu (s, Xs) ds + fr (s, Xs) dhs, tel  Xt = Xo + fu (s, Xs) ds + fr (s, Xs) dhs, tel  The general probabilistic representation of Solution to the  above pole (0) is thus given as
Starting from the fully Nom-linear poles taking the form  (244 + 11.0 x 4 + 1/2 Tr (ToT 0 x 4) = F(., ., 4, 0 x 4, 0 x 4)  (4(T.) = g dotined on [0, 7] x Rd — (0)  on 180 page 2  - let i de time the diffusion process X in the domain  18d associated with Non-linear pole (0) given as  t Xt = Xo + fu (s, Xs) ds + fr (s, Xs) dhs, tel  Xt = Xo + fu (s, Xs) ds + fr (s, Xs) dhs, tel  The general probabilistic representation of Solution to the  above pole (0) is thus given as
Starting from the fully Nom- linear poles taking  the form  (2eu + M. D. 2u + 1/2 Tr (00 T D. 2u) = F(., ., u, D. 2u, D. 2u)  (u(T.) = g detined on [0, T] x Rd - 0  on Ho page 2  - let's detine the diffusion process X in the domain  Rd associated with Non-linear pole (0 given as  t  Xt = Xo + Ju(s, Xs) ds + Ju(s, Xs) dNs, text

-Revision Materials For a PDE Derivation (Proof)
Starting from the fully Nom- Vincent poles taking
$F(\cdot, u, D_2u, D_2u)$
[ Deu + M. Deu + 1/2 Tr (OOT DEU) = F(,, u, Deu, Deu)
(U(1)) = 1 on Red
- let's detent the diffusion process X in the domain  Per associated with Non-linear pole O given as
In a sociated with Non-linear pare
$X_t = X_0 + \int u(s, X_s) ds + \int \sigma(s, X_s) dN_s$ , te[0,T]
o o
- The general probabilistic representation of Solution to the
above pole () is thus given as
$Y_{t} = g(X_{\tau}) - \int f(s, X_{s}, Y_{s}, Z_{s}) ds - \int Z_{s} dW_{s} + \epsilon C_{s} dW_{s}$
t Page 7
1) Farman-Kac formula
=> By applying the Feynmann-Kac formula
1 ( ) I a comina U (t, X2) 12 a smoot
function, this probabilistic representation is direct
function, this probabilistic representation obtained by Ito's formula applied to U(
obtained by Itos
$X_t = X_0 + \int u(s, X_s) ds + \int \sigma(s, X_s) dN_s$ , to [0,T]
- The general probabilistic representation of Solution to the above pole () is thus given as
$V_t = g(X_\tau) - \int f(s, X_s, Y_s, Z_s) ds - \int_t Z_s dW_s \qquad t \in [0, 1]$
1
=) By applying the Feynman-Kac formula
of U(+ x,) and assuming U(t, Xt) is a smooth
function, this probabilistic representation is directly
obtained by Itôs formula applied to Mathe
The $U(t, X_t)$ , and assuming $U(t, X_t)$ is a smooth function, this probabilistic representation is directly obtained by Itô's formula applied to $U(\pm, X_t)$ and on the interval $t \in [0,T]$
=> \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \
$= \sum_{t=1}^{\infty} \mathcal{I}_{t} = \mathcal{I}_{t} (t, X_{t})^{T} \mathcal{D}_{x} \mathcal{U}_{t} (t, X_{t})$
=) Now taking the subdivision of the interval (t,T)
The
$T \in [t,T]$

-Revision Materials For a PDE Derivation (Proof) Xt= Xo+ fu(s, Xs) ds + fo(s, Xe) dNs, to[0,T) - The general probabilistic representation of Solution to the above pole @ is thus given as 7= g(x+)- SF(s, xs,7s,2s)ds- SZs.dws + EEOT) => By applying the Feynman-Kac formala The U(t, Xt), and assuming U(t, Xt) is a smooth function, this probabilistic representation is directly obtained by Itôs formula applied to U(t, XE) and on the interval telo,T)  $Z_t = \int (t, X_t)^T D_x U(t, X_t)$ =) Now, taking the Subdivision of the interval (t,T)  $\pi \in [t,T]$ = Taking the madulus of IT, given as T = SUP, At = SUP, tith ti => We then consider the Euler-Manyama discretization (Xi) i=0, ---, N, given as  $X_{i} = X_{o} + \sum_{j=0}^{i-1} u(t_{j}, X_{j}) \Delta t_{j} + \sum_{j=0}^{i-1} \sigma(t_{j}, X_{j}) \Delta W_{j}$ = Ean & is critical since when Xt can't be div simulated, we vely on the paths of ean c

Page 7
=> Taking the modulus of TT, given as  T = SUP, At = SUP, tith ti
=> We then consider the Fuller-Morrygama discretization
=> We then consider the Editer
(V)
i-1 (1 x) 21 = T(+: X;) DW; -0
$X_i = X_{i-1} + \sum_{j=1}^{n} A_j \int_{X_j} dt_j + \sum_{j=1}^{n} A$
$X_{i} = X_{o} + \sum_{j=0}^{i-1} \mathcal{U}(t_{j}, X_{j}) \Delta t_{j} + \sum_{j=0}^{i-1} \mathcal{T}(t_{j}, X_{j}) \Delta W_{j} - \emptyset$
= Ean & is critical since when Xt can't be directly
simulated, we vely on the paths at ean a data during
simulated, we very training data during
=> This we use to act as a training data during
machine learning setting
Tracking (EN. 11)
the state of the s
=> Now, we write the time discretization of the
probabilistic representation of It equation by part
probabilistic representation of It equality
iterating relations on the interval [to,T] in
iterating relations
2007
page 7
= Taking the madulus of T, given as T = SUP, At = SUP, tim ti
=> Taking the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{int} = t_i$ The Sup them Consider the Euler-Morringama discretization
=> Taking the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{int} t_i$ => We then consider the Euler-Morrygama discretization
=> Taking the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{int} = t_i$ => We then consider the Euler-Morrygama discretization
=> Taking the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{int} t_i$ => We then consider the Euler-Morrygama discretization
=> Taking the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} - t_i$ => We then consider the Euler-Morringama discretization  (Xi) i=0,, N, given as  X; = Xo + $\sum_{j=0}^{N-1} M(t_j, X_j) \Delta t_j + \sum_{j=0}^{N-1} J(t_j, X_j) \Delta W_j - 0$
=> Tawing the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{int} t_i$ T = SUP; $\Delta t = SUP$ , $t_{int} t_i$ Then consider the Euler-Morrygama discretization  (Xi) i=0,, N, given as  Xi = Xo + $\sum_{j=0}^{i-1} M(t_j, X_j) \Delta t_j + \sum_{j=0}^{i-1} J(t_j, X_j) \Delta W_j$ The constant of the second streets as t
=> Tawing the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then consider the Euler-Morrygama discretization  (Xi) i=0,, N, given as  Xi = X. + $\sum_{j=0}^{i-1} M(t_j, X_j) \Delta t_j + \sum_{j=0}^{i-1} J(t_j, X_j) \Delta W_j$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $\Delta t = SU$
=> Tawing the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then consider the Euler-Morrygama discretization  (Xi) i=0,, N, given as  Xi = X. + $\sum_{j=0}^{i-1} M(t_j, X_j) \Delta t_j + \sum_{j=0}^{i-1} J(t_j, X_j) \Delta W_j$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{i+1} = t_i$ Then the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $\Delta t = SU$
=> Taking the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{int} t_i$ T = SUP; $\Delta t = SUP$ , $t_{int} t_i$ => We then consider the Euler-Morringama discretization  (Xi)i=0,, N, gran as  Xi = X. + $\sum_{j=0}^{N} M(t_j, X_j) \Delta t_j + \sum_{j=0}^{N-1} J(t_j, X_j) \Delta W_j - 0$ This we use to act as a training data during
=> Taking the malulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{int} t_i$ => We then consider the Euler-Morringama discretization  (Xi)i=0,, N, gran as  Xi = X. + $\sum_{j=0}^{N} M(t_j, X_j) \Delta t_j + \sum_{j=0}^{N-1} J(t_j, X_j) \Delta W_j - 0$ => Eqn & is evitical since when $X_t$ can't be divertly  simulated, we vely on the paths of eqn &  This we use to act as a training data during
=> Taking the madulus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{int} t_i$ => We then consider the Euler-Morringana discretization  (Xi)i=0,, N, given as  Xi = Xo + $\sum_{j=0}^{N} M(t_j, X_j) \Delta t_j + \sum_{j=0}^{N-1} J(t_j, X_j) \Delta W_j - Q$ => Ean & is critical since when $X_t$ can't be directly simulated, we rely on the paths of ean $Q$ Simulated, we rely on the paths of ean $Q$ This we use to act as a training data during machine learning setting
=> Taking the malulus of T, given as  T = SUP; At = SUP, time ti  T = SUP; At = SUP, time ti  Exceptions  (Xi)i=0,, N, given as  Xi = Xo + Z M (tj, Xj) Atj + Z  J=0  T (tj, Xj) ANj - Q  Xi = Xo + Z M (tj, Xj) Atj + Z  J=0  T (tj, Xj) ANj - Q  Xi = Xo + Z M (tj, Xj) Atj + Z  J=0  T (tj, Xj) ANj - Q  Xi = Xo + Z M (tj, Xj) Atj + Z  J=0  T (tj, Xj) ANj - Q  Xi = Xo + Z M (tj, Xj) Atj + Z  J=0  T (tj, Xj) ANj - Q  Xi = Xo + Z M (tj, Xj) Atj + Z  J=0  T (tj, Xj) ANj - Q  Xi = Xo + Z M (tj, Xj) Atj + Z  J=0  T (tj, Xj) ANj - Q  T (tj, Xj) ANj -
=> Taking the madelus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{int} = t_i$ => We then consider the Euler-Morringana discretization  (Xi)i=0,, N, given as  X; = X • + $\sum_{j=0}^{N} M(t_j, X_j) \Delta t_j + \sum_{j=0}^{N} J(t_j, X_j) \Delta W_j$ => Eqn & is without since when $X_t$ can't be directly  simulated, we vely another paths of earl &  => This we use to act as a training data during  machine learning setting  => Now, we write the time discretization of the
=> Taking the madelus of T, given as  T = SUP; $\Delta t = SUP$ , $t_{int} = t_i$ => We then consider the Euler-Morringana discretization  (Xi)i=0,, N, given as  X; = X • + $\sum_{j=0}^{N} M(t_j, X_j) \Delta t_j + \sum_{j=0}^{N} J(t_j, X_j) \Delta W_j$ => Eqn & is without since when $X_t$ can't be directly  simulated, we vely another paths of earl &  => This we use to act as a training data during  machine learning setting  => Now, we write the time discretization of the
=> Taking the malulus of TT, given as SUP, $\Delta_t = SUP$ , $t_{int} = t_i$ => We then consider the Eular-Manuagama descretization  (Xi)i=0,, N, given as  Xi = Xo + \( \frac{1}{2} \text{M}(t_i, X_i) \text{St}_i + \frac{1}{2} \text{T}(t_i, X_i) \text{AW}_i - \text{D}  Xi = Xo + \( \frac{1}{2} \text{M}(t_i, X_i) \text{St}_i + \frac{1}{2} \text{T}(t_i, X_i) \text{AW}_i - \text{D}  => Ean & is critical since when Xt carif be divertly simulated, we rely on the paths of ean & all and a training data during  => This we use to act as a training data during machine learning setting  => Now, we write the time discretization of the probabilistic representation of Yt equation by patoming iterating relations on the interval [to,T] in
=> Taking the malulus of T, given as SUP, $\Delta_t = SUP$ , $t_{int} t_i$ => We then consider the Eular-Manungama discretization  (Xi)i=0,, N, given as  I = X to the since when Xt can't be directly  Simulated, we rely on the paths of ean a simulated, we rely on the paths of ean a data during  This we use to act as a training data during machine learning setting  => Now, we write the time discretization of the probabilistic representation of Yt equation by patoming iterating relations on the interval [to,T] in
=> Tawing the malwin of T, given as SUP; $\Delta t = SUP$ , $t_{int} t_i$ => We then consider the Euler-Monograma decoretization  (Xi)=0,, N, given as  Xi = X • + $\sum_{j=0}^{N} U(t_j, X_j) \Delta t_j + \sum_{j=0}^{N} U(t_j, X_j) \Delta W_j$ => Ean & is without since when $X_t$ can't be divertly simulated, we vely so the paths of ean & simulated, we vely so the paths of ean & at as a training data during machine learning setting  => Now, we use to act as a training data during machine learning setting  => Now, we write the time discretization of the probabilistic representation of $\sum_{j=0}^{N} V_j = \sum_{j=0}^{N} V_j $
=> Tawing the malwin of T, given as SUP; $\Delta t = SUP$ , $t_{int} t_i$ => We then consider the Euler-Monograma decoretization  (Xi)=0,, N, given as  Xi = X • + $\sum_{j=0}^{N} U(t_j, X_j) \Delta t_j + \sum_{j=0}^{N} U(t_j, X_j) \Delta W_j$ => Ean & is without since when $X_t$ can't be divertly simulated, we vely so the paths of ean & simulated, we vely so the paths of ean & at as a training data during machine learning setting  => Now, we use to act as a training data during machine learning setting  => Now, we write the time discretization of the probabilistic representation of $\sum_{j=0}^{N} V_j = \sum_{j=0}^{N} V_j $
=> Tawing the malwin of T, given as SUP; $\Delta t = SUP$ , $t_{int} t_i$ => We then consider the Euler-Monograma decoretization  (Xi)=0,, N, given as  Xi = X • + $\sum_{j=0}^{N} U(t_j, X_j) \Delta t_j + \sum_{j=0}^{N} U(t_j, X_j) \Delta W_j$ => Ean & is without since when $X_t$ can't be divertly simulated, we vely so the paths of ean & simulated, we vely so the paths of ean & at as a training data during machine learning setting  => Now, we use to act as a training data during machine learning setting  => Now, we write the time discretization of the probabilistic representation of $\sum_{j=0}^{N} V_j = \sum_{j=0}^{N} V_j $
=> Taking the malulus of T, given as SUP, $\Delta_t = SUP$ , $t_{int} t_i$ => We then consider the Eular-Manungama discretization  (Xi)i=0,, N, given as  I = X to the since when Xt can't be directly  Simulated, we rely on the paths of ean a simulated, we rely on the paths of ean a data during  This we use to act as a training data during machine learning setting  => Now, we write the time discretization of the probabilistic representation of Yt equation by patoming iterating relations on the interval [to,T] in
= Tawn, the malulus of T, given as  TT = Sup, $\Delta t = Sup$ , $time t$ ;  => We then consider the Euler-Mondyama discretization  (Xi) = 0,, o, given as  Xi = X + 1 \(\frac{1}{2}\) \(\text{M}(t_j, X_j) \(\Delta t_j + \frac{1}{2}\) \(\text{U}(t_j, X_j) \(\Delta t_j + \frac{1}{2}\) \(\Delta t_j + \Delta t_j
=> Tawing the malwin of T, given as SUP; $\Delta t = SUP$ , $t_{int} t_i$ => We then consider the Euler-Monograma decoretization  (Xi)=0,, N, given as  Xi = X • + $\sum_{j=0}^{N} U(t_j, X_j) \Delta t_j + \sum_{j=0}^{N} U(t_j, X_j) \Delta W_j$ => Ean & is without since when $X_t$ can't be divertly simulated, we vely so the paths of ean & simulated, we vely so the paths of ean & at as a training data during machine learning setting  => Now, we use to act as a training data during machine learning setting  => Now, we write the time discretization of the probabilistic representation of $\sum_{j=0}^{N} V_j = \sum_{j=0}^{N} V_j $

-Revision Materials For a PDE Derivation (Proof) Taking the madulus of 11, given to Sup, At = Sup, tite to => We then consider the Euler-Manyama discretization (Xi)i=0, --, N, given as  $X_i = X_{\bullet} + \sum_{j=0}^{i-1} u(t_j, X_j) \Delta t_j + \sum_{j=0}^{i-1} \sigma(t_j, X_j) \Delta W_j = \emptyset$ =) Ean & is critical since when Xt can't be directly simulated, we vely on the paths of ean & almang data during machine learning setting => Now, we write the time discretization of the probabilistic representation of Yt equation by partoning terrating relations on the interval [to,T] in backward induction or Y, = E; [ Y, +1 - f (+1, X; , Y, = 2) A+; EI[ AW; YT , 1=0, -- N-1 where E; denotes the fti conditional expectation of Assuming we have a terminal relation, the equi (an he written (turmulated) it evatively | X: = X. + \( \sigma \text{if } (t\_j, X\_j) \( \sigma t\_j + \sigma \) \( \text{if } \sigma t\_j \) =) Ean & is without since when Xt could be directly simulated, we vely on the paths of ean & during =) This we use to act as a training data during machine learning setting => Now, we write the time discretization of the probabilistic representation at It equation by partoning iterating relations on the interval [to,T] in backward induction as = E; [7, x, X, X, Z, Z, Dt; Z:= E([ AN: YTT , 1=0, -- N-1 where E; denotes the ft: conditional expectation of Assuming we have a terminal relation, the equilibrium Can be written (terminated) it evatively 7" = g(xn) - \( \sum\_{i=1}^{\text{T}} \left(t; \text{X}\_j, \text{T}\_j, \text{Z}\_j^\mathbb{T}, \text{Z}\_j^\mathbb{T}) \text{Dt} + \text{Z}\_j^\mathbb{T} \text{DW}\_j To prove this process, we start by starting

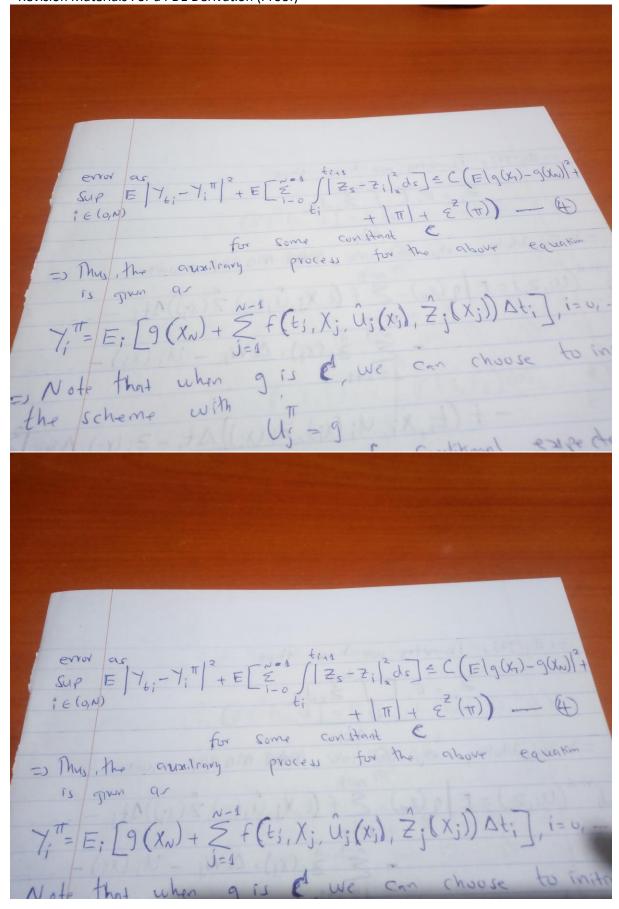
-Revision Materials For a PDE Derivation (Proof) => This we use to act as machine learning setting => Now, we write the time discretization at the probabilistic representation at It equation by partoming iterating relations on the interval [to,T] in backward induction or 7, = E; [7, +1 - f (ti, Xi, Yi, Zi) st; Zi = EI [ AWi YT , i=0, -- N-1 where E; denotes the fti conditional expectation

The equ (3)

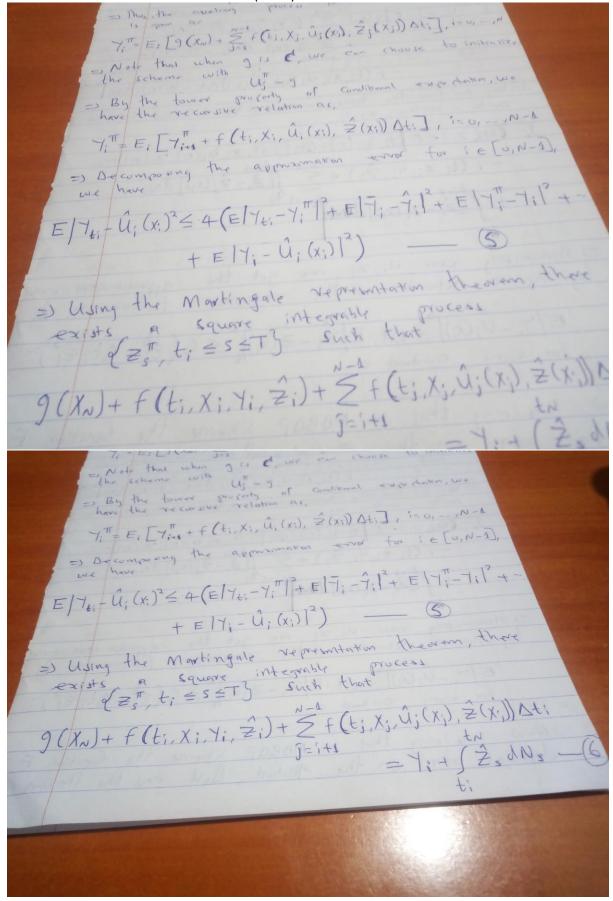
Can be written (formulated) iteratively as  $7! = g(x_N) - \sum_{j=1}^{N-1} [f(t_i, X_j, Y_j^T, Z_j^T) \Delta t + Z_j^T, \Delta W_j]_{i=0,-}$ =) to prove this process, we start by staring that TN = 9 (XN), then we define the time discret  $X_i = X_* + \stackrel{i-1}{\leq} M(t_i, X_i) \Delta t_j + \stackrel{i-1}{\leq} \sigma(t_i, X_j) \Delta W_i$ => Enn & is without since when Xt can't be directly of simulated, we vely another parties of ean & during => This we use to act as a training data during machine (earning setting => Now, we write the time discretization of the probabilistic representation of the equation by partoning iterating relations on the interval [to,T] in backward induction or 7 = E: [7" - f (ti, Xi, Yi, Zi) Dti Z:= E([ AN; Y" , i=0, -- N-1 where E; denotes the ft; conditional expectation

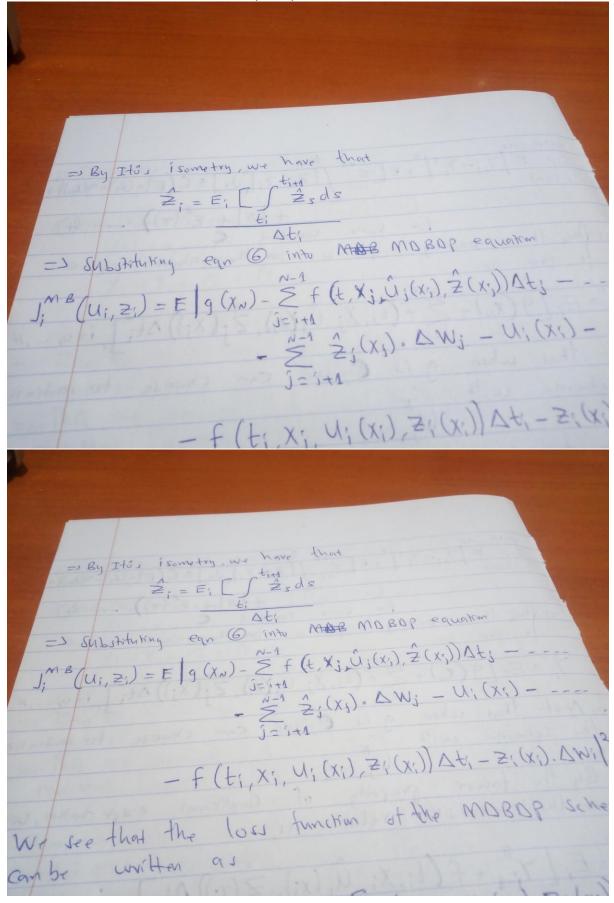
The early

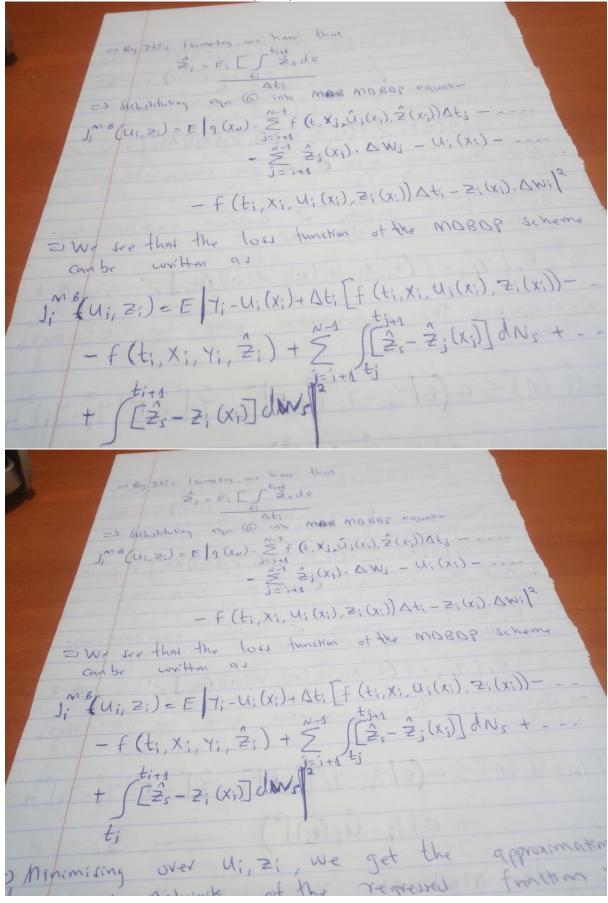
Can be written (formulated) iteratively as 71 = g (xn) - E [f(t:,Xj,,Yj,,Zj) Dt + Zj. DWj]:=0.-N = To prove this process, we short by 'Antry that TN = 9 (XN), then we define the time discretization

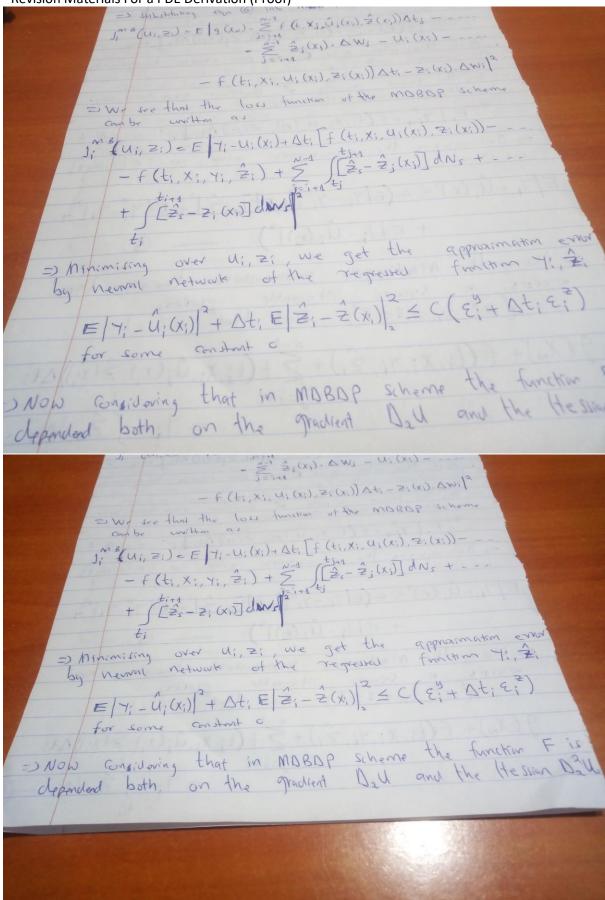


-Revision Waterials For a FDE Derivation (F1001)
12
error as 13 = ~ 1 1 2 1 = C (E/g(x)-g(xn))+
SUP E YLI-YIT + E Z / Zs-tila
error as $Sup =  Y_{t_{i}} - Y_{i}^{T} ^{2} + E[\sum_{i=0}^{N-1} \int_{t_{i}}^{t_{i+1}}  Z_{s} - Z_{i} _{2}^{2} ds] = C(E g(X_{t}) - g(X_{N}) ^{2} +  T_{s}  +  T_{s}  +  Z_{s} ^{2} (T_{s}) - (T_{s})$
=> Thus, the auxiliary process for the above equation  13 Thus ar
the society for the aport of
=> Thus, the auxiliary
Thus, the auxiliary  1s gran as $Y_i^{T} = E : \left[ g(x_N) + \sum_{j=1}^{N-1} f(t_j, X_j, \hat{U}_j(x_j), \hat{Z}_j(x_j)) \Delta t_i \right], i = 0,, N$ $Y_i^{T} = E : \left[ g(x_N) + \sum_{j=1}^{N-1} f(t_j, X_j, \hat{U}_j(x_j), \hat{Z}_j(x_j)) \Delta t_i \right]$ Can choose to initialize
- T - [a(x)+ 5 f(t), X; U; (x), +j(x))
7 = E: Ly (NN) + Z=1
I all & we can (hoose
- Note that when
the scheme with
The scheme with the tower property of Conditional expectation, we
n. he tower property of constituent
= By the relation as
the scheme with Us = 9  By the tower property of conditional expectation, we have the recursive relation as,
νανε της της + f (t; , X; , Û; (x;), ≥(x;)) Δt;], i=0,-, Λ Υ; = Ε; [Υ;+++ f (t; , X; , Û; (x;), ≥(x;)) Δt;], i=0,-, Λ
T - [ + X; U; (Xi), Z(Xi)) 4th 3
7: = E: 17:+1 + T ( ) ( ) ( )
maker eva for le [o
maker Erra 10, 106
error as $ Y_{i} - Y_{i}^{T} ^{2} + E[z^{-1}]^{2}  z_{s} - z_{i} ^{2} ds = C(E q(x_{1}) - q(x_{N}) ^{2} +  T  + e^{z}(T)) - E$ if $(q_{N})$ constant
error as [ ] + E = [ ]   Zs - Zi   ds ] = C(E)
Sup E 76; - 1;   + E I o d; +   TT   + E (TT) - (T)  i E (a, N)  for some constant  process for the above equation  rs gram as  rs gram as
for some constant the above equation
- Thus the auxiliary process to
Thus, the auxiliary process por $X_i = \sum_{j=1}^{N-1} \left[ \frac{1}{2} \left( \frac{1}{2} \right) \left( \frac{1}$
- TO(x)+ Sf(ts, X; U; (x)), Z; (x))
7; = E: [] (NN) J=1
Male that when g is & we
the scheme with it
21 = 3 conditional expectation, we
- Bu the tower property of
7, t = E: [9 (XN) + \( \) f (ts, Xj, Uj (Xj), Zj \( \) Xj \\ \) Yole that when g is \( \) we can choose to initralize the scheme with \( \) Uj = 9 \\  By the tower property of conditional expectation, we have the recursive relation as,
have the reconside relation of YiT = E: [Yits + f(t; Xi, Û; (xi), Z(xi)) Ati], i=0,-,N-1
$\pi = [X']$ , $f(X)$ , $f(X)$ , $f(X)$
1; # Li L its
h maken ever tor le love
- Lecompound the approximation
=) Decomposing the approximation evol for le[0,N-1] we have
1 2 (1 )
-14 11 (x) < 4(E) /4-1; 1+41
17/4 - U; (x:) = 4(E/Y+:-Y;     + E/T; -Y:   + E/T; -T:
1 1 (x) 12 (x)









-Revision Materials For a PDE Derivation (Proof) - f (ti, xi, U; (xi), Z((xi)) Dt, - Z((xi). Dwi)2 ZI We see that the loss function of the MOBOR scher Ji (ui, Zi) = E /1; -u; (xi) + At; [f (ti, xi, u; (xi), Zi (xi)) - $-f(t_{1}, x_{1}, y_{1}, \hat{z}_{1}) + \sum_{j=i+1}^{N-1} \left[\hat{z}_{s} - \hat{z}_{j}(x_{j})\right] dN_{s} + \int_{z_{s}}^{z_{j+1}} \left[\hat{z}_{s} -$ =) Minimising over U; Zi we get the approximation evolution of the regressed function 1:, Zi [ = | Y; - U; (x;) | + Dt; E | Z; - Z(x;) | 2 < C(E; + Dt; E; =)

for some constant c JOW Considering that in MBBDP scheme the function F is depended both on the gradient Dall and the Hessian Dall and with the assumption that the general solution U to the pole is smooth, then we shall dende our minimized approximation error solution over Y, Z, T valued in IRX IRX Sd by  $Y_t = U(t, X_t)$   $Z_t = \Delta_{\alpha}U(t, X_t)$   $\Gamma_t = D_{\alpha}^2U(t, X_t)$ over the interval to [0,T] Using the above hypothesis procedure and applying the Ito's formula to It function, then the end applying approximation of (Y, Z, P) satisfying the bar

